Sun Java™ System

Portal Server Secure Remote Access 6
관리 설명서

2004Q2

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

부품 번호 : 817-7145

목차

그림 목록 **11**

표 목록 **13**

절차 목록 **15**

본 설명서에 대해 **21**

이 설명서의 독자 21
주지해야 할 사항 22
이 설명서의 구성 22
이 설명서에 사용된 문서 약속 24
고정 폭 글꼴 24
기울임꼴 글꼴 24
대괄호 25
명령줄 프롬프트 25
관련 정보 출처 25
관련된 타사 웹 사이트 참조 26
본 설명서의 온라인 버전 26

1 장 Portal Server Secure Remote Access 소개 **27**
SRA 소프트웨어 개요 27
열린 모드 28
보안 모드 29
SRA 서비스 31
게이트웨이 31
Rewriter 31
NetFile 32
Netlet 32

Proxylet 32
SRA 제품 관리 32
SRA 속성 구성 33
충돌 해결 설정 34
지원 응용 프로그램 35

2 장 게이트웨이 37
게이트웨이의 개요 38
게이트웨이 프로필 만들기 38
platform.conf 파일 이해 40
chroot 환경에서 게이트웨이 실행 47
chroot 환경에서 게이트웨이 다시 시작 49
게이트웨이의 다중 인스턴스 만들기 50
홈이 여러 개인 게이트웨이 인스턴스 만들기 52
같은 LDAP 를 사용하여 게이트웨이 인스턴스 만들기 52
게이트웨이 시작 및 중지 53
게이트웨이 다시 시작 54
가상 호스트 지정 55
Identity Server 에 접속하도록 프록시 지정 56
웹 프록시 사용 56
자동 프록시 구성 사용 62
Netlet 프록시 사용 65
Netlet 프록시의 인스턴스 만들기 68
Netlet 프록시 사용 설정 69
Netlet 프록시 다시 시작 69
Rewriter 프록시 사용 70
Rewriter 프록시의 인스턴스 만들기 70
Rewriter 프록시 사용 설정 72
Rewriter 프록시 다시 시작 72
게이트웨이에서 역 프록시 사용 73
클라이언트 정보 가져오기 73
인증 체이닝 사용 76
와일드카드 인증 사용 77
브라우저 캐싱 사용 해제 77
게이트웨이 서비스 사용자 인터페이스 사용자 정의 78
연합 관리 사용 79
연합 관리 시나리오 79
연합 관리 리소스 구성 80

3 장 Proxylet 및 Rewriter 87
Proxylet 개요 88
Proxylet 사용의 이점 88

Proxylet 구성 89
Rewriter 개요 89
문자 집합 암호화 90
Rewriter 사용 시나리오 90
URLScraper 91
게이트웨이 91
규칙 집합 작성 92
공용 인터페이스 (규칙 집합 DTD) 92
예제 XML DTD 95
규칙 작성을 위한 절차 97
규칙 집합 가이드라인 97
규칙 집합의 루트 요소 정의 98
재귀적 기능 사용 99
언어 기반 규칙 정의 (규칙 정의) 99
HTML 컨텐트에 대한 규칙 99
JavaScript 컨텐트에 대한 규칙 106
XML 컨텐트에 대한 규칙 121
CCS(Cascading Style Sheet) 에 대한 규칙 124
WML 에 대한 규칙 125
재귀적 기능 사용 125
게이트웨이 서비스에서 Rewriter 구성 125
기본 작업 126
고급 작업 131
디버깅 로그 사용의 문제 해결 135
Rewriter 디버깅 수준 설정 135
디버깅 파일 이름 136
작업 예제 137
HTML 컨텐트 예제 139
JavaScript 컨텐트 예제 148
XML 속성 예제 168
사례 연구 170
6.x 규칙 집합을 3.0 과 매핑 175

4 장 NetFile 177
NetFile 개요 177
지원되는 파일 액세스 프로토콜 178
NetFile 에 대한 액세스 활성화 179
NetFile 의 로깅 활성화 180
UNIX 인증 구성 180

5 장 Netlet 183
Netlet 개요 183
Netlet 구성 요소 184
Netlet 사용 시나리오 186
Netlet 작업 186
원격 호스트에서 애플릿 다운로드 187
Netlet 규칙 정의 187
규칙의 유형 191
Netlet 규칙 예제 194
샘플 Netlet 규칙 199
Netlet 로깅 사용 설정 202
Sun Ray 환경에서 Netlet 실행 203
새로운 HTML 파일 203
Deprecated HTML 파일 : 205

6 장 PDC 가 있는 Netlet 207
PDC 를 위한 Netlet 구성 207

7 장 인증서 209
SSL 인증서의 개요 210
인증서 파일 210
인증서 트러스트 속성 211
CA 트러스트 속성 212
certadmin 스크립트 216
직접 서명한 인증서 생성 217
인증서 서명 요청 (CSR) 생성 219
루트 CA 인증서 추가 221
인증 기관으로부터 SSL 인증서 설치 222
CA 로부터 인증서 주문 222
CA 로부터 받은 인증서 설치 223
인증서 삭제 224
인증서의 트러스트 속성 수정 226
루트 CA 인증서 나열 227
모든 인증서 나열 229
인증서 인쇄 230

8 장 URL 액세스 제어 구성 233
거부된 URL 목록 설정 234
허용된 URL 목록 설정 235
단일 사인온 관리 235

9 장 게이트웨이 구성 239
핵심 탭 240
HTTP 및 HTTPS 연결 사용 241
Rewriter 프록시 목록 사용과 만들기 241
Netlet 활성화 243
Netlet 프록시 목록 활성화 및 만들기 244
Proxylet 활성화 245
쿠키 관리 활성화 245
HTTP 기본 인증 사용 246
HTTP 지속 연결 사용 247
지속 연결당 최대 요청 수 지정 248
지속성 소켓 연결에 시간 초과 지정 249
반환 시간을 위한 계정의 유예 시간 초과 지정 249
사용자 세션 쿠키를 URL 로 전달 목록 만들기 250
최대 연결 대기 길이 지정 251
게이트웨이 시간 초과 지정 252
최대 스레드 풀 크기 지정 253
캐시된 소켓 시간 초과 지정 253
Portal Server 목록 만들기 254
서버 재시도 간격 지정 255
외부 서버 쿠키 저장 활성화 255
URL 에서 세션 얻기 256
쿠키를 안전하다고 표시 257
프록시 탭 257
웹 프록시 사용 활성화 258
웹 프록시에 대한 URL 목록 만들기 258
사용하지 않을 프록시의 URL 목록 만들기 259
도메인 및 부속 도메인의 프록시 목록 만들기 260
프록시 비밀번호 목록 만들기 261
자동 프록시 구성 지원 사용 261
자동 프록시 구성 파일 위치 지정 262
웹 프록시를 통한 Netlet 터널링 활성화 263
보안 탭 263
인증되지 않은 URL 목록 만들기 264
인증서 사용 가능 게이트웨이 호스트 목록 만들기 264
40 비트 암호화 연결 허용 265
SSL 버전 2.0 사용 266
SSL 암호 선택 사용 266
SSL 버전 3.0 사용 267
Null 암호 활성화 268
신뢰할 수 있는 SSL 도메인 목록 만들기 268
개인 디지털 인증서 (PDC) 인증 구성 269

Rewriter 탭 272
모든 URL 의 다시 쓰기 사용 272
규칙 집합에 대한 URI 의 매핑 목록 만들기 273
구문 분석할 MIME 유형 목록 만들기 276
기본 도메인 및 부속 도메인 지정 277
다시 쓰지 않을 URI 목록 만들기 278
MIME 추측 활성화 278
구문 분석할 URI 매핑 목록 만들기 279
마스크 활성화 280
마스킹용 씨드 문자열 지정 280
마스크하지 않을 URI 목록 만들기 281
게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 만들기 282
기록 탭 282
기록 사용 283
Netlet 기록 사용 284

10 장 NetFile 구성 285
호스트 탭 286
OS 문자 집합 지정 286
호스트 검색 순서 지정 287
공통 호스트 목록 구성 288
기본 도메인 지정 290
Windows 도메인 / 워크그룹 지정 291
기본 WINS/DNS 서버 지정 292
다른 유형의 호스트에 액세스 지정 293
허용된 호스트 목록 구성 294
거부된 호스트 목록 구성 295
권한 탭 296
보기 탭 298
NetFile 창 크기 지정 298
NetFile 창 위치 지정 299
작업 탭 300
임시 파일 디렉토리 지정 300
파일 업로드 제한 크기 설정 301
디렉토리 검색 제한 지정 302
압축 지정 303
일반 탭 304
MIME 유형 구성 파일 위치 지정 304
NetFile 에 디버깅 활성화 305

11 장 Netlet 구성 307
사용자에 Netlet 서비스 지정 308
Netlet 규칙 추가 309
기존 Netlet 규칙 수정 310
Netlet 규칙 삭제 311
기본 암호 지정 311
기본 루프백 포트 할당 312
연결에 대한 재인증 활성화 313
연결에 대한 경고 팝업 대화 상자 활성화 313
포트 경고 대화 상자에 확인란 표시 활성화 314
연결 유지 시간 315
포털 로그아웃할 때 Netlet 종료 옵션 설정 315
Netlet 규칙에 대한 액세스 정의 316
Netlet 규칙에 액세스 거부 317
호스트에 대한 액세스 허용 318
호스트에 대한 액세스 거부 318
프록시 구성 319
디버그 로깅 활성화 320

12 장 Proxylet 구성 321
Proxylet 구성 321

13 장 SSL 가속기 구성 323
개요 323
Sun Crypto Accelerator 1000 323
Crypto Accelerator 1000 사용 324
Crypto Accelerator 1000 구성 324
Sun Crypto Accelerator 4000 327
Crypto Accelerator 4000 사용 327
Crypto Accelerator 4000 구성 328
외부 SSL 장치 및 프록시 가속기 330
외부 SSL 장치 가속기 사용 330
외부 SSL 장치 가속기 구성 331

부록 A 로그 파일 333

부록 B 구성 속성 335
액세스 목록 서비스 335
게이트웨이 서비스 336
핵심 336
프록시 338

보안 339
Rewriter 340
기록 343
NetFile 서비스 343
호스트 343
권한 345
보기 345
작업 345
일반 347
Netlet 서비스 347
Proxylet 서비스 349

부록 C 국가 코드 351

용어 361

그림 목록

그림 1-1 열린 모드에서 Portal Server 29
그림 1-2 보안 모드에 있는 Portal Server(SRA 소프트웨어 사용) 30
그림 2-1 웹 프록시 관리 57
그림 2-2 Netlet 프록시 구현 67
그림 5-1 Netlet 구성 요소 184

표 목록

표 2-1 platform.conf 파일 속성 41
표 2-2 [도메인 및 부속 도메인의 프록시] 목록에서 항목 매핑 60
표 2-3 HTTP 헤더의 정보 74
표 3-1 * 와일드카드 사용의 실례 106
표 3-2 Rewriter 디버깅 파일 136
표 3-3 예제 규칙 집합과 사례 연구 사이의 매핑 173
표 3-4 SP3 의 규칙 매핑 175
표 4-1 파일 시스템 및 지원되는 프로토콜 178
표 5-3 예제 Netlet 규칙 200
표 7-1 인증서 파일 211
표 7-2 인증서 트러스트 속성 212
표 7-3 공인 인증 기관 212
표 13-1 Crypto Accelerator 1000 설치 점검 목록 324

절차 목록

충돌 해결 수준을 설정하려면 34
게이트웨이 프로필을 만들려면 39
chroot 를 설치하려면 47
chroot 환경에서 게이트웨이를 다시 시작하려면 49
게이트웨이를 시작하려면 53
게이트웨이를 중지하려면 54
다른 프로필로 게이트웨이를 다시 시작하려면 54
게이트웨이를 다시 시작하려면 54
게이트웨이 워치독을 구성하려면 55
가상 호스트를 지정하려면 55
프록시를 지정하려면 56
Netlet 프록시를 다시 시작하려면 69
Netlet 프록시 워치독을 구성하려면 70
Rewriter 프록시를 다시 시작하려면 72
Rewriter 프록시 워치독을 구성하려면 72
역 프록시를 활성화하려면 73
기존 PDC 인스턴스에 인증 모듈을 추가하려면 76
브라우저 캐싱을 사용 해제하려면 77
게이트웨이가 모든 URL 을 다시 쓰도록 하려면 126
URI 를 규칙 집합에 매핑하려면 128
MIME 매핑을 지정하려면 129
다시 쓰지 않을 URI 를 지정하려면 129
기본 도메인을 지정하려면 130
MIME 추측을 사용하려면 131
URI 매핑을 구문 분석하려면 132
마스크를 활성화하려면 132
마스크 씨드 문자열을 지정하려면 133

마스크하지 않을 URI 목록을 지정하려면 134
게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 하려면 134
Rewriter 디버깅 수준을 설정하려면 135
HTML 속성 예제를 사용하려면 139
HTML JavaScript 토큰 예제를 사용하려면 141
폼 예제를 사용하려면 144
애플릿 예제를 사용하려면 146
JavaScript URL 변수 예제를 사용하려면 148
JavaScript Expression 변수 예제를 사용하려면 151
JavaScript DHTML 변수 예제를 사용하려면 153
JavaScript DJS 변수 예제를 사용하려면 156
JavaScript System 변수 예제를 사용하려면 158
JavaScript URL 함수 예제를 사용하려면 160
JavaScript Expressions 함수 예제를 사용하려면 162
JavaScript DHTML 함수 예제를 사용하려면 164
JavaScript DJS 함수 예제를 사용하려면 166
XML 속성 예제를 사용하려면 168
OWA 규칙 집합을 구성하려면 174
조직 및 사용자에 대해 NetFile 을 활성화하려면 179
Unix 인증을 사용하려면 180
Unix 인증을 구성하려면 180
규칙을 추가한 후 Netlet 을 실행하려면 198
PDC 를 위한 Netlet 을 구성하려면 207
설치 후 직접 서명한 인증서를 생성하려면 217
CSR 을 생성하려면 219
루트 CA 인증서를 추가하려면 221
CA 로부터 인증서를 주문하려면 222
CA 로부터 받은 인증서를 설치하려면 223
인증서를 삭제하려면 224
인증서의 트러스트 속성을 수정하려면 226
루트 CA 목록을 보려면 227
모든 인증서를 나열하려면 229
인증서를 인쇄하려면 230
URL 액세스 제어를 구성하려면 233
거부된 URL 목록을 설정하려면 234
허용된 URL 목록을 설정하려면 235
호스트의 단일 사인온을 비활성화하려면 236

세션별 단일 사인온을 활성화하려면 236
인증 수준을 지정하려면 237
게이트웨이 속성을 구성하려면 239
HTTP 또는 HTTPS 모드에서 실행되도록 게이트웨이를 구성하려면 241
Rewriter 프록시를 사용하고 목록을 만들려면 242
Netlet 을 사용하려면 243
Netlet 프록시를 사용하고 목록을 만들려면 244
Proxylet 을 활성화하려면 245
쿠키 관리를 사용하려면 246
HTTP 기본 인증을 사용하려면 247
HTTP 지속 연결을 사용하려면 247
지속 연결당 최대 요청 수를 지정하려면 248
지속성 소켓 연결의 시간 초과를 지정하려면 249
반환 시간을 위한 계정의 유예 시간 초과를 지정하려면 249
쿠키 URL 전달을 추가하려면 251
최대 연결 대기 길이를 지정하려면 251
게이트웨이 시간 초과를 지정하려면 252
최대 스레드 풀 크기를 지정하려면 253
캐시된 소켓 시간 초과를 지정하려면 253
포털 서버를 지정하려면 254
포털 서버 재시도 간격을 지정하려면 255
외부 서버 쿠키를 저장하려면 255
URL 에서 세션을 얻으려면 256
쿠키를 안전하다고 표시하려면 257
웹 프록시의 사용을 설정하려면 258
웹 프록시에 대한 URL 을 지정하려면 258
사용하지 않을 URL 을 지정하려면 259
도메인 및 부속 도메인의 프록시를 지정하려면 260
프록시 비밀번호를 지정하려면 261
자동 프록시 구성 지원을 사용하려면 262
자동 프록시 구성 파일 위치를 지정하려면 262
웹 프록시를 통한 터널 Netlet 을 사용하려면 263
인증되지 않은 URL 경로를 지정하려면 264
게이트웨이에 인증서 사용 가능 호스트 목록을 추가하려면 264
40 비트 암호화 연결을 허용하려면 265
SSL 버전 2.0 을 사용하려면 266
개별 암호 선택을 사용하려면 266

SSL 버전 3.0 을 사용하려면 267
Null 암호를 활성화하려면 268
인증된 SSL 도메인 목록을 만들려면 268
PDC 및 코드화된 장치를 구성하려면 269
게이트웨이가 모든 URL 을 다시 쓰도록 하려면 272
URI 를 규칙 집합에 매핑하려면 274
OWA 규칙 집합을 구성하려면 275
MIME 매핑을 지정하려면 276
기본 도메인 및 부속 도메인을 지정하려면 277
기본 도메인 및 부속 도메인을 지정하려면 278
MIME 추측을 사용하려면 278
URI 매핑을 구문 분석하려면 279
마스크를 활성화하려면 280
마스킹용 씨드 문자열을 지정하려면 280
마스크하지 않을 URI 를 지정하려면 281
게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 하려면 282
게이트웨이 기록을 사용하려면 283
Netlet 기록을 사용하려면 284
NetFile 속성을 구성하려면 285
OS 문자 집합을 지정하려면 286
호스트 검색 순서를 지정하려면 287
공통 호스트 목록을 구성하려면 289
기본 도메인을 지정하려면 290
기본 Windows 도메인 또는 워크그룹을 지정하려면 291
기본 WINS/DNS 서버를 지정하려면 292
다른 유형의 호스트에 액세스를 지정하려면 293
허용된 호스트 목록을 만들려면 294
거부된 호스트 목록을 만들려면 295
권한을 사용 / 사용 해제하려면 297
NetFile 창의 크기를 지정하려면 298
NetFile 창의 위치를 지정하려면 299
임시 디렉토리를 지정하려면 300
파일 업로드 제한 크기를 설정하려면 301
디렉토리 검색 제한을 지정하려면 302
기본 압축 유형을 지정하려면 303
MIME 유형 구성 파일의 위치를 지정하려면 304
Netlet 규칙을 추가하려면 309

Netlet 규칙을 수정하려면 310
Netlet 규칙을 삭제하려면 311
기본 암호를 지정하려면 312
기본 루프백 포트를 할당하려면 312
연결에 대한 재인증 활성화하려면 313
연결에 대한 경고 팝업을 비활성화하려면 313
사용자가 포트 경고 대화 상자를 나타나지 않도록 허용하려면 314
연결 유지 시간을 설정하려면 315
[포털 로그아웃할 때 Netlet 종료] 옵션을 설정하려면 315
Netlet 규칙에 대한 액세스를 정의하려면 316
Netlet 규칙에 액세스를 거부하려면 317
호스트에 액세스를 허용하려면 318
호스트에 액세스를 거부하려면 318
Proxylet 속성을 구성하려면 321
Crypto Accelerator 1000 을 구성하려면 324
Crypto Accelerator 4000 을 구성하려면 328
외부 SSL 장치 가속기를 구성하려면 331

본 설명서에 대해

이 설명서에서는 Sun Java™ System Portal Server Secure Remote Access (이전의 Sun ONE™ Portal Server, Secure Remote Access) 를 관리하는 방법을 설명합니다 .

Sun Java System Portal Server Secure Remote Access (SRA) 는 원격 사용자가 인터넷을 통해 해당 조직의 네트워크와 그 서비스에 안전하게 액세스할 수 있도록 합니다 . 그 외에도 조직에 안전한 인터넷 포털을 제공하여 특정 대상 (직원 , 비즈니스 파트너 또는 일반 대중) 이 컨텐트 , 응용 프로그램 및 데이터에 액세스하도록 할 수 있습니다 .

SRA 는 Solaris™ 8.0 및 9.0 운영 체제에서 실행됩니다 . 이 설명서에는 SRA 의 구성 및 관리에 관한 지침도 들어 있습니다 .

이 서문은 다음 절로 구성됩니다 .

- 이 설명서의 독자
- 주지해야 할 사항
- 이 설명서의 구성
- 이 설명서에 사용된 문서 약속
- 관련 정보 출처
- 본 설명서의 온라인 버전

이 설명서의 독자

이 설명서에서는 독자가 UNIX® 시스템 및 TCP/IP 네트워크 관리 경험이 있는 네트워크 또는 시스템 관리자라고 가정합니다 . 이러한 독자가 SRA 의 설치 , 구성 및 관리를 담당합니다 .

SRA 의 다양한 구성 요소를 설치하기 위해 사용하는 컴퓨터에 루트 액세스가 필요합니다 . 사용자 및 서비스의 구성과 같이 기타 작업을 수행하기 위해 관리 권한도 필요합니다 .

주지해야 할 사항

SRA 을 관리하기 전에 다음 사항에 익숙해야 합니다 .

- 기본적인 Solaris 관리 절차
- LDAP
- Sun Java™ System Directory Server
- Sun Java™ System Web Server
- Sun Java™ System Portal Server

Rewriter 규칙을 작성하려면 다음도 필요합니다 .

- HTML 및 HTML 태그에 대한 이해
- JavaScript 에 대한 상당한 지식
- XML 에 대한 기본 지식

이 설명서의 구성

이 설명서는 다음 장과 부록으로 구성되어 있습니다 .

본 설명서에 대해 (현재 장)

1 장 , “Portal Server Secure Remote Access 소개 ”

이 장에서는 SRA 에 대해 설명하며 , Sun Java™ System Portal Server 제품과 SRA 구성 요소 간의 관계에 대해서도 설명합니다 . SRA 의 관리와 구성에 대한 정보도 제공합니다 .

2 장 " 게이트웨이 "

이 장에서는 게이트웨이 관련 개념과 게이트웨이의 원활한 실행에 필요한 정보를 설명합니다 .

3 장 "Proxylet 및 Rewriter"

이 장에서는 Proxylet 과 Rewriter 에 대해 설명합니다 . Rewriter 의 경우에는 샘플 규칙과 모범 사례도 제공합니다 .

4 장 "NetFile"

이 장에서는 NetFile 과 그 작업에 대해 설명합니다 .

5 장 "Netlet"

이 장에서는 Netlet 을 사용하여 사용자의 원격 표준 포털 데스크탑과 인트라넷에서 응용 프로그램을 실행하는 서버 사이에서 응용 프로그램을 안전하게 실행하는 방법을 설명합니다 .

6 장 "PDC 가 있는 Netlet"

이 장에서는 Netlet 을 PDC 에서 사용할 수 있도록 클라이언트 브라우저의 Java 플러그인을 구성하는 방법을 설명합니다 .

7 장 " 인증서 "

이 장에서는 인증서 관리를 설명하고 직접 서명한 인증서 또는 인증 기관에서 받은 인증서를 설치하는 방법을 알아봅니다 .

8 장 "URL 액세스 제어 구성 "

이 장에서는 특정 URL 에 대한 게이트웨이를 통해 최종 사용자에 대한 액세스를 허용 또는 거부하는 방법을 설명합니다 .

9 장 " 게이트웨이 구성 "

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 게이트웨이 속성을 구성하는 방법에 대해 설명합니다 .

10 장 "NetFile 구성 "

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 NetFile 을 구성하는 방법에 대해 설명합니다 .

11 장 "Netlet 구성 "

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 Netlet 속성을 구성하는 방법에 대해 설명합니다 .

12 장 "Proxylet 구성 "

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 Proxylet 을 구성하는 방법에 대해 설명합니다 .

13 장 "SSL 가속기 구성 "

이 장에서는 Portal Server Secure Remote Access 에 다양한 가속기를 구성하는 방법을 설명합니다 .

부록 A, " 로그 파일 "

이 부록에는 Portal Server Secure Remote Access 로그 파일과 관련 설명이 모두 나열되어 있습니다 .

부록 B, " 구성 속성 "

이 부록에는 Sun Java™ System Identity Server 관리 콘솔에서 Portal Server Secure Remote Access 에 설정하는 속성이 나열되어 있습니다 .

부록 C, " 국가 코드 "

이 부록에는 인증서 관리 중에 지정해야 하는 2 자로 된 국가 코드가 나열되어 있습니다 .

용어

이 용어집에는 Sun Java System 의 전체 용어집으로 가는 링크가 나와 있습니다 .

이 설명서에 사용된 문서 약속

고정 폭 글꼴

고정 폭 글꼴은 컴퓨터 화면에 나타나는 텍스트나 사용자가 입력해야 하는 텍스트에 사용됩니다 . 파일 이름 , 구분 이름 , 함수 및 예제에도 사용됩니다 .

기울임꼴 글꼴

*기울임꼴 글꼴*은 해당 설치에 고유한 정보를 사용하여 사용자가 입력하는 텍스트 (예 : 변수) 를 나타낼 때 사용됩니다 . 서버 경로 , 이름 및 계정 아이디에 사용됩니다 .

대괄호

대괄호 [] 는 선택사항 매개 변수를 둘러쌀 때 사용됩니다 . 예를 들어 , 본 문서에서 xx 명령의 사용을 다음과 같이 설명하는 것을 볼 수 있습니다 .

```
xx [options] [action] [component]
```

[options], [arguments] 및 [component] 는 xx 명령에 추가할 수 있는 옵션 매개 변수가 있다는 것을 나타냅니다 .

명령줄 프롬프트

명령줄 프롬프트 (예를 들어 , C-Shell 의 경우 %, Korn 이나 Bourne shell 의 경우 $) 는 예제에 표시되지 않습니다 . 사용하고 있는 운영 체제 환경에 따라 다양한 명령줄 프롬프트를 보게 됩니다 . 그러나 특별한 언급이 없는 한 명령을 문서에 나타나는 대로 입력해야 합니다 .

관련 정보 출처

SRA 문서 자료

아래에는 추가적인 SRA 문서를 나열했습니다 .

- *Portal Server Secure Remote Access 속성 온라인 도움말*
- *Portal Server Secure Remote Access Netlet 온라인 (클라이언트) 도움말*
- *Portal Server Secure Remote Access NetFile Java1 온라인 (클라이언트) 도움말*
- *Portal Server Secure Remote Access NetFile Java2 온라인 (클라이언트) 도움말*
- *Portal Server Secure Remote Access Proxylet 온라인 (클라이언트) 도움말*

Portal Server 문서

Sun Java™ System Portal Server 문서 모음에는 다음이 포함됩니다 .

- *Portal Server 관리 설명서*
- *Portal Server Migration Guide*
- *Portal Server Desktop Customization Guide*
- *Portal Server Developer's Guide*

본 설명서에서 참조한 문서

본 설명서에서 참조한 기타 문서 :

- *Identity Server 관리 설명서*
- *Sun Crypto Accelerator 1000 Board Installation and User ' s Guide*

  본 설명서는 http://www.sun.com/products-n-solutions/hardware/docs/pdf/816-2450-11.pdf에서 찾아볼 수 있습니다 .

관련된 타사 웹 사이트 참조

http://docs.sun.com 에서 Sun 기술 문서를 온라인으로 이용할 수 있습니다 . 아카이브를 찾아보거나 특정 문서 제목이나 주제를 검색할 수 있습니다 .

참고 Sun 은 본 문서에서 언급하는 타사 웹 사이트를 이용할 수 있는지에 대해 보장하지 않습니다 . Sun 은 이러한 사이트나 리소스를 통해 이용할 수 있는 어떤 컨텐트 , 광고 , 제품 또는 기타 자료에 대해 책임지거나 이를 승인하지 않습니다 . Sun 은 이러한 사이트나 리소스를 통해 이용할 수 있는 컨텐트 , 재화 또는 서비스를 사용하거나 이에 의존한 결과로 또는 이와 연관하여 발생하는 모든 실제적 또는 주장되는 손해나 손상에 대해 책임지지 않습니다 .

본 설명서의 온라인 버전

웹 사이트 http://docs.sun.com 에서 Sun 기술 문서를 온라인으로 이용할 수 있습니다 . 아카이브를 찾아보거나 특정 문서 제목이나 주제를 검색할 수 있습니다 .

Portal Server Secure Remote Access 소개

이 장에서는 Sun Java™ System Portal Server Secure Remote Access (이전의 Sun ONE™ Portal Server, Secure Remote Access) 에 대해 설명하며 , Sun Java™ System Portal Server (Portal Server) 소프트웨어와 Sun Java System Portal Server Secure Remote Access (SRA) 구성 요소 간의 관계에 대해서도 설명합니다 .

이번 장에서는 다음 주제를 다룹니다 .

- SRA 소프트웨어 개요
- SRA 서비스
- SRA 제품 관리
- SRA 속성 구성
- 지원 응용 프로그램

SRA 소프트웨어 개요

SRA 소프트웨어는 원격 사용자가 인터넷을 통해 해당 조직의 네트워크와 그 서비스에 안전하게 액세스할 수 있도록 합니다 . 그 외에도 조직에 안전한 인터넷 포털을 제공하여 특정 대상 (직원 , 비즈니스 파트너 또는 일반 대중) 이 컨텐트 , 응용 프로그램 및 데이터에 액세스 할 수 있게 해 줍니다 .

SRA 소프트웨어는 어떤 원격 장치에서도 브라우저를 통해 포털 컨텐트와 서비스에 원격으로 안전하게 액세스하도록 합니다. Secure Remote Access 는 저렴하고 안전한 액세스 솔루션으로서 Java 기술이 구현된 브라우저가 있는 어떤 장치에서도 사용자가 액세스할 수 있어 클라이언트 소프트웨어의 필요성을 없애줍니다. Portal Server 와의 통합으로 사용자는 액세스 권한을 가지고 있는 컨텐트와 서비스에 암호화된 안전한 액세스를 하게 됩니다.

SRA 소프트웨어는 매우 안전한 원격 액세스 포털을 구축하는 기업에 이상적입니다. 이러한 포털은 보안, 안전 및 인트라넷 리소스의 기밀 유지에 중점을 둡니다. SRA 구조는 이런 유형의 포털에 잘 적합합니다. SRA 소프트웨어에서는 사용자가 인터넷에 자원을 노출하지 않고도 인터넷을 통해 인트라넷 자원에 안전하게 액세스할 수 있습니다.

비무장 지대 (DMZ) 에 위치하는 게이트웨이는 모든 인트라넷 URL, 파일 시스템 및 응용 프로그램에 대한 단일한 보안 액세스 포인트를 제공합니다. 세션, 인증 및 표준 포털 데스크탑과 같은 SRA 이외의 기타 모든 서비스는 DMZ 뒤의 안전한 인트라넷에 상주합니다. 클라이언트 브라우저에서 게이트웨이로의 통신은 HTTPS 를 사용하여 암호화됩니다. 게이트웨이에서 서버 및 인트라넷 자원으로의 통신은 HTTP 또는 HTTPS 일 수 있습니다.

SRA 소프트웨어는 두 가지 방법을 사용합니다.

Netlet 및 NetFile 애플릿이 클라이언트 컴퓨터로 다운로드되고 지원 파일은 게이트웨이나 Portal Server 호스트에 위치할 수 있습니다.

Portal Server 는 두 모드로 작동할 수 있습니다.

- 열린 모드
- 보안 모드

열린 모드

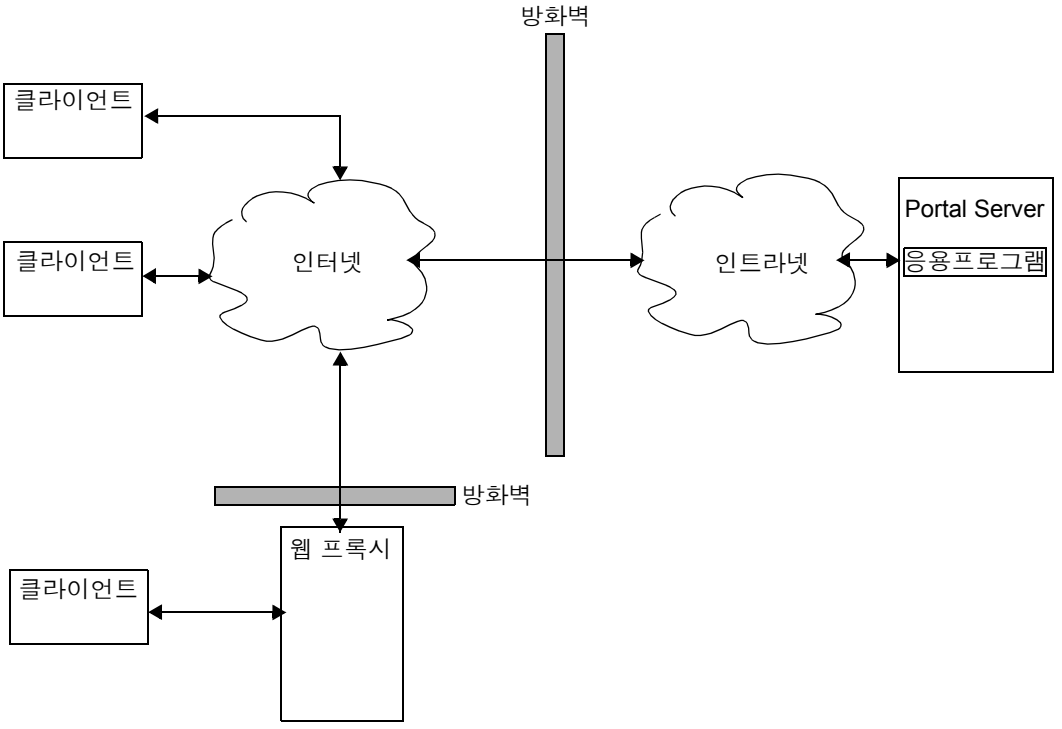
열린 모드에서 Portal Server 는 SRA 소프트웨어 없이 설치됩니다. 이 모드에서 HTTPS 통신이 가능하지만 안전한 원격 액세스는 불가능합니다. 즉, 사용자는 원격 파일 시스템과 응용프로그램에 안전하게 액세스할 수 없습니다.

개방 포털과 보안 포털 사이의 주된 차이점은 개방 포털로 제공되는 서비스가 일반적으로 안전한 인트라넷 내부가 아닌 비무장 지대 (DMZ) 에 위치한다는 것입니다. DMZ 는 공용 인터넷과 사설 인트라넷 사이의 작은 보호 네트워크로서 일반적으로 양쪽에서 방화벽으로 경계를 이룹니다.

포털에 중요한 정보가 들어있지 않다면 (공용 정보를 배포하고 무료 응용프로그램에 액세스 허용) 이 모드를 사용하여 보안 모드를 사용할 때보다 대량의 사용자가 제출하는 액세스 요청에 보다 빠르게 응답할 수 있습니다 .

그림 1-1 은 열린 모드에 있는 Portal Server 를 보여줍니다 . 여기서 Portal Server 는 방화벽 뒤의 단일 서버에 설치됩니다 . 여러 클라이언트가 단일 방화벽을 통해 인터넷에서 Portal Server 에 액세스합니다 .

그림 1-1 열린 모드에서 Portal Server



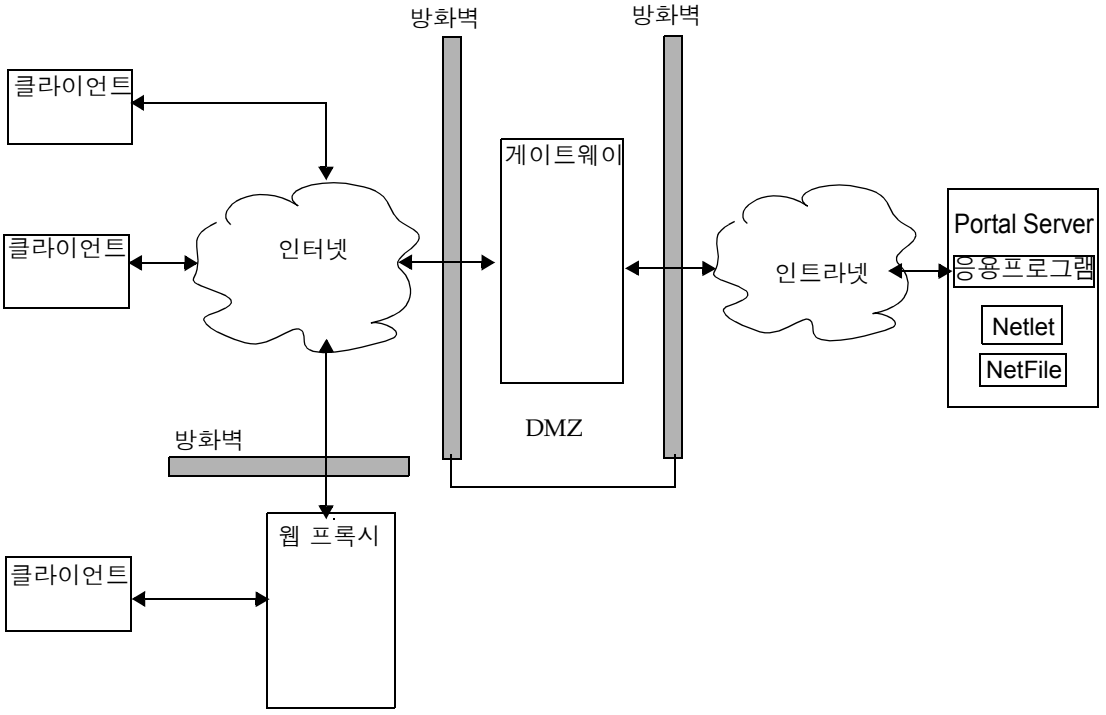
보안 모드

보안 모드에서 사용자는 필요한 인트라넷 파일 시스템과 응용프로그램에 안전하게 원격으로 액세스할 수 있습니다 .

게이트웨이는 비무장 지대 (DMZ) 에 상주합니다 . 게이트웨이는 모든 인트라넷 URL 과 응용 프로그램에 단일한 보안 액세스 포인트를 제공하여 방화벽에서 열린 포트의 수를 줄입니다 . 세션 , 인증 및 표준 포털 데스크탑과 같은 기타 모든 Portal Server 서비스는 DMZ 뒤의 안전한 인트라넷에 상주합니다 . 클라이언트 브라우저에서 게이트웨이로의 통신은 SSL(Secure Sockets Layer) 상에서 HTTP 를 사용하여 암호화됩니다 . 게이트웨이에서 서버 및 인트라넷 리소스로의 통신은 HTTP 또는 HTTPS 일 수 있습니다 .

그림 1-2 는 SRA 소프트웨어가 사용된 Portal Server 를 보여줍니다 . SSL 은 인터넷에서 클라이언트와 게이트웨이 사이의 연결을 암호화하는 데 사용됩니다 . SSL 은 게이트웨이와 서버 사이의 연결을 암호화할 때도 사용됩니다 . 인트라넷과 인터넷 사이에 게이트웨이가 있어 클라이언트와 Portal Server 간의 보안 경로가 확장됩니다 .

그림 1-2 보안 모드에 있는 Portal Server(SRA 소프트웨어 사용)



추가 서버와 게이트웨이를 추가하여 사이트를 확장할 수 있습니다 . SRA 소프트웨어는 비즈니스 요구 사항에 따라 다양한 방법으로 구성할 수 있습니다 .

SRA 서비스

SRA 소프트웨어에는 5 가지 주요 구성 요소가 있습니다 .

- 게이트웨이
- Rewriter
- NetFile
- Netlet
- Proxylet

게이트웨이

SRA 게이트웨이는 인터넷을 통해 들어오는 원격 사용자 세션과 회사 인트라넷 사이에서 인터페이스와 보안 장벽을 제공합니다 . 게이트웨이는 원격 사용자에 대한 단일 인터페이스를 통해 내부 웹 서버와 응용 프로그램 서버에서 안전하게 컨텐트를 제공합니다 .

웹 서버는 HTML, JavaScript 및 XML 과 같은 웹 기반 자원을 사용하여 클라이언트와 게이트웨이 사이에서 통신합니다 . Rewriter 는 웹 컨텐트를 이용할 수 있도록 만드는 데 사용되는 게이트웨이 구성 요소입니다 .

응용 프로그램 서버는 텔넷 및 FTP 와 같은 이진 프로토콜을 사용하여 클라이언트와 게이트웨이 사이에서 통신합니다 . 게이트웨이에 상주하는 Netlet 이 이 목적으로 사용됩니다 . 자세한 내용은 2 장 " 게이트웨이 " 를 참조하십시오 .

Rewriter

Rewriter 는 최종 사용자가 인트라넷을 둘러보고 이 페이지의 링크와 기타 URL 참조가 제대로 작동하도록 합니다 . Rewriter 는 웹 브라우저의 위치 필드에 게이트웨이 URL 을 붙여 컨텐트 요청을 게이트웨이를 통해 리디렉션합니다 . 자세한 내용은 3 장 "Proxylet 및 Rewriter" 를 참조하십시오 .

NetFile

NetFile 은 파일 시스템과 디렉토리에 원격으로 액세스하여 작업할 수 있도록 하는 파일 관리자 응용 프로그램입니다 . NetFile 에는 Java 기반의 사용자 인터페이스인 NetFile Java™ 가 포함되어 있습니다 . Java 1 및 Java 2 를 이용할 수 있습니다 . 자세한 내용은 4 장 "NetFile" 을 참조하십시오 .

Netlet

Netlet 은 원격 데스크탑에서 일반 또는 회사별 응용 프로그램을 안전하게 실행하도록 지원합니다 . 해당 사이트에 Netlet 을 구현하면 사용자가 Telnet 및 SMTP 와 같은 일반적 TCP/IP 서비스와 pcANYWHERE 또는 Lotus Notes 같은 HTTP 기반 응용프로그램을 안전하게 실행할 수 있습니다 . 자세한 내용은 5 장 "Netlet" 를 참조하십시오 .

Proxylet

Proxylet 은 클라이언트 컴퓨터에서 실행되는 동적 프록시 서버입니다 . Proxylet 은 URL 을 게이트웨이로 리디렉션합니다 . 이 작업은 클라이언트 시스템의 브라우저에서 프록시 설정을 읽은 다음 로컬 프록시 서버 또는 Proxylet 을 가리키도록 수정하는 방식으로 이루어집니다 .

SRA 제품 관리

SRA 소프트웨어에는 관리를 위한 두 가지 인터페이스가 있습니다 .

- Identity Server 관리 콘솔
- 명령줄

대부분의 관리 작업은 웹 기반의 Sun Java System Identity Server 관리 콘솔을 통해 이루어집니다 . 관리 콘솔은 로컬로 또는 웹 브라우저를 통해 원격으로 액세스할 수 있습니다 . 그러나 , 파일 수정과 같은 작업은 UNIX 명령줄 인터페이스를 통해 관리해야 합니다 .

SRA 속성 구성

조직 , 역할 및 사용자 수준에서 SRA 에 관련된 속성을 구성할 수 있습니다 . 단 , 다음의 예외가 있습니다 .

- 충돌 해결 수준은 사용자 수준에서 설정할 수 없습니다 . [서비스 구성] 탭에서도 이용할 수 없습니다 . 34 페이지의 " 충돌 해결 설정 " 을 참조하십시오 .
- MIME 유형 구성 파일 위치 속성은 조직 수준에서만 설정할 수 있습니다 . 304 페이지의 "MIME 유형 구성 파일 위치 지정 " 을 참조하십시오 .

조직 수준에서 설정된 값은 이 조직의 모든 규칙과 사용자에 상속됩니다 . 사용자 수준에서 설정된 값은 조직 또는 규칙 수준에서 설정된 값보다 우선합니다 .

대부분의 속성은 Identity Server 의 [Identity Server] 탭 또는 [서비스 구성] 탭에서 설정할 수 있습니다 . 서비스 구성 수준에서 설정된 속성은 템플리트로 쓰입니다 . 생성되는 새로운 조직 또는 사용자는 기본적으로 이 값을 상속합니다 .

서비스 구성 수준에서 속성 값을 변경할 수 있습니다 . 이러한 새로운 값은 새 조직이 추가될 때만 반영됩니다 . [서비스 구성] 탭에서 속성 값을 변경해도 기존 조직이나 사용자에게는 영향을 미치지 않습니다 . 자세한 내용은 *Identity Server 관리 설명서*를 참조하십시오 .

다음 서비스를 사용하여 SRA 구성 아래의 Identity Server 관리 콘솔에서 SRA 속성을 구성합니다 .

- 액세스 목록

  이 서비스를 통해 특정 URL 에 대한 액세스를 허용 또는 제한하고 단일 사인온 기능을 관리할 수 있습니다 . 자세한 내용은 8 장 "URL 액세스 제어 구성 " 을 참조하십시오 .

- 게이트웨이

  이 서비스를 사용하여 프록시 관리 , 쿠키 관리 , 로깅 , Rewriter 관리 및 암호와 같은 모든 게이트웨이 관련 속성을 구성할 수 있습니다 . 자세한 내용은 9 장 " 게이트웨이 구성 " 을 참조하십시오 .

- NetFile

  이 서비스를 사용하여 공통 호스트 , MIME 유형 및 여러 호스트 유형에 대한 액세스 등 모든 NetFile 관련 속성을 구성할 수 있습니다 . 자세한 내용은 10 장 "NetFile 구성 " 을 참조하십시오 .

- Netlet

  이 서비스를 사용하여 Netlet 규칙 , 필요한 규칙에 대한 액세스 , 조직 및 호스트 그리고 기본 알고리즘과 같은 모든 Netlet 관련 속성을 구성할 수 있습니다 . 자세한 내용은 11 장 "Netlet 구성 " 을 참조하십시오 .

- Proxylet

  이 서비스를 사용하면 Proxylet 애플릿 바인딩 IP 주소 및 포트 번호와 같은 Proxylet 관련 속성을 구성할 수 있습니다 . 자세한 내용은 12 장 "Proxylet 구성 " 을 참조하십시오 .

주의 게이트웨이는 게이트웨이가 실행되는 동안 이루어지는 속성 변경에 대해 알림을 받지 않습니다 .

게이트웨이에서 업데이트된 프로필 속성 (게이트웨이나 다른 서비스에 속하는) 을 사용하도록 게이트웨이를 다시 시작합니다 . 76 페이지의 " 인증 체이닝 사용 " 을 참조하십시오 .

충돌 해결 설정

➤ **충돌 해결 수준을 설정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 해당 서비스 (액세스 목록 , NetFile 또는 Netlet) 옆에 있는 화살표를 클릭합니다 .
7. [충돌 해결 수준] 필드 드롭다운 목록에서 필요한 수준을 선택합니다 .
8. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

지원 응용 프로그램

SRA 소프트웨어는 다음 응용프로그램을 지원합니다 .

- SRA 소프트웨어는 MS Exchange 2000 SP3 설치와 Outlook Web Access (OWA) 의 MS Exchange 2003 을 지원합니다 .

  OWA 페이지에 필요한 규칙 집합은 `exchange_2003_owa_ruleset` 이라는 이름으로 바로 설치됩니다 . OWA 에 대한 사례 연구를 보려면 275 페이지의 "Outlook Web Access 의 규칙 집합 " 을 참조하십시오 .
- iNotes - Notes 5.0.11
- 캘린더 - Sun ONE Calendar Server 릴리스 5.1.1 이상
- Messenger Express - iPlanet Messaging Server 5.2 이상

2 장

게이트웨이

이 장에서는 게이트웨이 관련 개념과 게이트웨이의 원활한 실행에 필요한 정보를 설명합니다. 게이트웨이 구성에 대한 자세한 내용은 9 장 " 게이트웨이 구성 " 을 참조하십시오.

이번 장에서는 다음 주제를 다룹니다.

- 게이트웨이의 개요
- 게이트웨이 프로필 만들기
- platform.conf 파일 이해
- chroot 환경에서 게이트웨이 실행
- chroot 환경에서 게이트웨이 다시 시작
- 게이트웨이의 다중 인스턴스 만들기
- 게이트웨이 시작 및 중지
- 게이트웨이 다시 시작
- 가상 호스트 지정
- Identity Server 에 접속하도록 프록시 지정
- 웹 프록시 사용
- 자동 프록시 구성 사용
- Netlet 프록시 사용
- Rewriter 프록시 사용
- 게이트웨이에서 역 프록시 사용
- 클라이언트 정보 가져오기

- 인증 체이닝 사용
- 와일드카드 인증 사용
- 브라우저 캐싱 사용 해제
- 게이트웨이 서비스 사용자 인터페이스 사용자 정의
- 연합 관리 사용

게이트웨이의 개요

게이트웨이는 인터넷을 통해 들어오는 원격 사용자 세션과 회사 인트라넷 사이에서 인터페이스와 보안 장벽을 제공합니다 . 게이트웨이는 원격 사용자에 대한 단일 인터페이스를 통해 내부 웹 서버와 응용프로그램 서버에서 안전하게 컨텐트를 제공합니다 .

게이트웨이 프로필 만들기

게이트웨이 프로필에는 게이트웨이가 수신하는 포트 , SSL 옵션 및 프록시 옵션과 같이 게이트웨이 구성에 관련된 모든 정보가 들어 있습니다 .

게이트웨이를 설치할 때 기본값을 선택하면 " 기본 " 이라는 기본 게이트웨이 프로필이 만들어집니다 . 기본 프로필에 해당하는 구성 파일은 다음 위치에 있습니다 .

`/etc/opt/SUNWps/platform.conf.default`

여기서 `/etc/opt/SUNWps` 는 모든 `platform.conf.*` 파일을 위한 기본 위치입니다 .

`platform.conf` 파일 내용에 대한 자세한 내용은 40 페이지의 "platform.conf 파일 이해 " 를 참조하십시오 .

가능한 작업

- 여러 프로필을 만들어 각 프로필에 대한 속성을 정의한 다음 이 프로필을 필요에 따라 서로 다른 게이트웨이에 할당할 수 있습니다 .
- 서로 다른 컴퓨터에 있는 게이트웨이 설치에 단일 프로필을 할당할 수 있습니다.

- 같은 컴퓨터에서 실행되는 단일 게이트웨이 인스턴스에 서로 다른 프로필을 할당할 수 있습니다 .

주의 같은 컴퓨터에서 실행되는 게이트웨이의 서로 다른 인스턴스에 같은 프로필을 할당하지 마십시오 . 그러면 포트 번호가 같게 되므로 충돌이 발생합니다 .

같은 게이트웨이에 만들어진 서로 다른 프로필에서 같은 포트 번호를 지정하지 마십시오 . 동일한 게이트웨이의 포트 번호가 같은 다중 인스턴스를 실행하면 충돌이 발생합니다 .

➤ 게이트웨이 프로필을 만들려면

1. Sun Java™ System Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 게이트웨이 페이지가 표시됩니다 .
4. [새로 만들기] 를 누릅니다 .

   새 게이트웨이 프로필 만들기 페이지가 표시됩니다 .
5. 새 게이트웨이 프로필의 이름을 입력합니다 .
6. 드롭다운 목록에서 새 프로필을 만들 때 사용할 프로필을 선택합니다 .

   기본적으로 만들어지는 새 프로파일은 모두 사전 제공된 기본 프로파일을 기준으로 합니다 . 사용자 정의 프로필을 만든 경우 드롭다운 목록에서 그 프로필을 선택할 수 있습니다 . 새 프로필은 선택한 프로필의 모든 속성을 상속합니다 .
7. [만들기] 를 누릅니다 .

   새 프로필이 만들어지며 새 프로필이 나열된 게이트웨이 페이지로 돌아갑니다 .
8. 변경 사항을 적용하려면 이 게이트웨이 프로필 이름이 있는 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

게이트웨이를 구성하려면 9 장 " 게이트웨이 구성 " 을 참조하십시오 .

platform.conf 파일 이해

platform.conf 파일은 기본적으로 다음 위치에 있습니다 .

/etc/opt/SUNWps

platform.conf 파일에는 게이트웨이에 필요한 상세 정보가 들어 있습니다 . 이 부분에서는 예제 platform.conf 파일이 나와 있으며 모든 항목에 대해 설명합니다 .

모든 컴퓨터별 상세 정보를 구성 파일에 포함시키면 좋은 점은 공통 프로필을 여러 컴퓨터에서 실행되는 게이트웨이에서 공유할 수 있다는 것입니다 .

다음은 예제입니다 .

```
#
# Copyright 11/28/00 Sun Microsystems, Inc. All Rights Reserved.
# "@(#)platform.conf1.38 00/11/28 Sun Microsystems"
#
gateway.user=noaccess
gateway.jdk.dir=/usr/java_1.3.1_06
gateway.dsame.agent=http://pserv2.iportal.com:8080/sunportal/RemoteConfigS
ervlet
portal.server.protocol=http
portal.server.host=pserv2.iportal.com
portal.server.port=8080
gateway.protocol=https
gateway.host=siroe.india.sun.com
gateway.port=333
gateway.trust_all_server_certs=true
gateway.trust_all_server_cert_domains=false
gateway.virtualhost=siroe1.india.sun.com 10.13.147.81
gateway.virtualhost.defaultOrg=o=root,dc=test,dc=com
gateway.notification.url=/notification
gateway.retries=6
gateway.debug=error
```

```
gateway.debug.dir=/var/opt/SUNWps/debug
gateway.logdelimiter=&&
gateway.external.ip=10.12.147.71
gateway.certdir=/etc/opt/SUNWps/cert/portal
gateway.allow.client.caching=true
gateway.userProfile.cacheSize=1024
gateway.userProfile.cacheSleepTime=60000
gateway.userProfile.cacheCleanupTime=300000
gateway.bindipaddress=10.12.147.71
gateway.sockretries=3
gateway.enable.accelerator=false
gateway.enable.customurl=false
gateway.httpurl=http://siroe.india.sun.com
gateway.httpsurl=https://siroe.india.sun.com
gateway.favicon=https://siroe.india.sun.com
gateway.logging.password=ALKJDF123SFLKJJSDFU
portal.server.instance=
gateway.cdm.cacheSleepTime
gateway.cdm.cacheCleanUpTime
```

표 2-1에는 platform.conf 파일에 있는 모든 필드가 나열되고 이에 대한 설명이 나와 있습니다 .

표 2-1 platform.conf 파일 속성

항목	기본값	설명
gateway.user	noaccess	게이트웨이가 이 사용자로 실행됩니다 . 게이트웨이는 루트로 시작되어야 하며 초기화 후에는 이 사용자가 되는 루트 권한을 상실합니다 .
gateway.jdk.dir		게이트웨이에서 사용하는 JDK 디렉토리의 위치입니다 .
gateway.dsame.agent		이 프로필을 얻을 수 있도록 시작하는 중에 게이트웨이에서 접촉하게 되는 Identity Server 의 URL 입니다 .

표 2-1 platform.conf 파일 속성

항목	기본값	설명
portal.server.protocol portal.server.host portal.server.port		기본 Portal Server 설치에서 사용하는 프로토콜, 호스트 및 포트입니다.
gateway.protocol gateway.host gateway.port		게이트웨이 프로토콜, 호스트 및 포트입니다. 이 값은 설치 시 지정한 모드 및 포트와 동일합니다. 이 값은 알림 URL 을 구성하는 데 사용됩니다.
gateway.trust_all_server_certs	true	게이트웨이에서 모든 서버 인증서를 신뢰해야 하는지 아니면 게이트웨이 인증서 데이터베이스에 있는 서버 인증서만 신뢰해야 하는지 나타냅니다.
gateway.trust_all_server_cert_domains	false	게이트웨이와 서버 사이에 SSL 통신이 있을 때마다 서버 인증서가 게이트웨이에 제공됩니다. 기본적으로 게이트웨이는 서버 호스트 이름이 서버 인증서 CN 과 같은지 확인합니다. 이 속성 값이 true 로 설정되어 있으면 게이트웨이에서는 수신하는 서버 인증서에 대해 도메인 확인을 사용하지 않습니다.
gateway.virtualhost		게이트웨이 컴퓨터에 구성된 호스트 이름이 여러 개 있을 경우 이 필드에서 이름을 다르게 지정하여 공급자 주소를 구분할 수 있습니다.

표 2-1 platform.conf 파일 속성

항목	기본값	설명
gateway.virtualhost.defaultOrg=org		사용자가 로그인할 기본 조직을 지정합니다 . 예를 들어 가상 호스트 필드 항목이 다음과 같다고 가정해 봅시다 . gateway.virtualhost=test.com employee.test.com Managers.test.com 기본 조직 항목이 다음과 같음 : test.com.defaultOrg = o=root,dc=test,dc=com employee.test.com.defaultOrg = o=employee,dc=test,dc=com Manager.test.com.defaultOrg = o=Manager,dc=test,dc=com 사용자는 https://manager.test.com 을 통해 https://test.com/o=Manager,dc=test,dc=com 대신 관리자 조직에 로그인할 수 있습니다. 참고 : virtualhost 및 defaultOrg 는 platform.conf file 에서는 대소문자가 구별되지만 URL 에 사용할 때는 구별되지 않습니다 .
gateway. notification.url		게이트웨이 호스트 , 프로토콜 및 포트 조합은 알림 URL 을 구성하는 데 사용됩니다 . 이 조합은 Identity Server 의 세션 알림을 수신하는 데 사용됩니다 . 알림 URL 은 다른 조직 이름과 같지 않도록 합니다 . 알림 URL 은 조직 이름과 일치하므로 해당 조직에 연결을 시도하는 사용자에게는 로그인 페이지 대신 공백 페이지가 나타납니다 .
gateway.retries		시작하는 중에 게이트웨이에서 Portal Server 에 접촉하고자 시도하는 횟수를 말합니다 .

표 2-1 platform.conf 파일 속성

항목	기본값	설명
gateway.debug	error	게이트웨이의 디버그 수준을 설정합니다 . 디버그 파일은 *debug-directory*/files에 있습니다 . 디버그 파일 위치는 gateway.debug.dir 항목에 지정되어 있습니다 . 디버깅 수준은 다음과 같습니다 . 오류 - 디버그 파일에 심각한 오류만 기록됩니다 . 일반적으로 이러한 오류가 발생하면 게이트웨이는 기능이 중단됩니다 . 경고 - 경고 메시지가 기록됩니다 . 매시지 - 모든 디버그 메시지가 기록됩니다 . 날짜 - 모든 디버그 메시지가 콘솔에 표시됩니다 . 디버그 파일은 다음과 같습니다 . srapGateway.*gateway-profile-name* - 게이트웨이 디버그 메시지가 들어 있습니다 . Gateway_to_from_server.*gateway-profile-name* - 메시지 모드에서는 이 파일에 게이트웨이와 내부 서버 사이의 모든 요청 및 응답 헤더가 들어 있습니다 . 이 파일을 생성하려면 /var/opt/SUNWps/debug 디렉토리에서 쓰기 권한을 변경합니다 . Gateway_to_from_server.*gateway-profile-name* - 메시지 모드에서는 이 파일에 게이트웨이와 클라이언트 브라우저 사이의 모든 요청 및 응답 헤더가 들어 있습니다 . 이 파일을 생성하려면 /var/opt/SUNWps/debug 디렉토리에서 쓰기 권한을 변경합니다 .
gateway.debug.dir		여기는 모든 디버그 파일이 생성되는 디렉토리입니다 . 이 디렉토리에는 gateway.user 에서 언급된 사용자가 파일에 쓸 수 있도록 충분한 권한을 가지고 있어야 합니다 .
gateway. logdelimiter		현재 사용되지 않음 .
gateway.external.ip		다중 홈 게이트웨이 컴퓨터인 경우 (IP 주소가 여러 개) 여기서 외부 IP 주소를 지정해야 합니다 . 이 IP 는 Netlet 에서 FTP 를 실행하는 데 사용됩니다 .

표 2-1 platform.conf 파일 속성

항목	기본값	설명
gateway.certdir		인증서 데이터베이스의 위치를 지정합니다 .
gateway.allow.client.caching	true	클라이언트 캐싱을 허용하거나 허용 불가합니다. 허용되는 경우 클라이언트 브라우저는 동적 페이지와 이미지를 캐싱하여 성능을 향상시킵니다 (네트워크 트래픽 감소를 통해). 허용되지 않는 경우 보안이 철저해서 클라이언트 쪽에서는 아무 것도 캐싱되지 않으므로 네트워크 로드가 많을 경우에는 성능 저하가 생깁니다 .
gateway.userProfile.cacheSize		게이트웨이에서 캐싱되는 사용자 프로필 항목의 수입니다 . 항목 수가 이 값을 초과하면 캐시를 정리하는 재시도가 자주 이루어집니다 .
gateway.userProfile.cacheSleepTime		초 단위로 캐시 정리에 대한 절전 시간을 설정합니다 .
gateway.userProfile.cacheCleanupTime		이 시간이 지나면 프로필 항목을 삭제할 수 있는 최대 시간 (초).
gateway.bindipaddress		다중 홈 컴퓨터에서 게이트웨이가 serversocket 을 바인딩하는 IP 주소입니다 . 모든 인터페이스를 청취하도록 게이트웨이를 구성하려면 gateway.bindipaddress=0.0.0.0 이 되도록 IP 주소를 변경합니다 .
gateway.sockretries	3	현재 사용되지 않음 .
gateway.enable.accelerator	false	true 로 설정된 경우 외부 가속기 지원이 허용됩니다 .
gateway.enable.customurl	false	true 로 설정된 경우 관리자는 게이트웨이에서 페이지를 다시 쓸 사용자 정의 URL 를 지정할 수 있습니다 .
gateway.httpurl		HTTP 역 프록시 URL 을 입력하여 게이트웨이에서 페이지를 다시 쓸 사용자 정의 URL 을 설정합니다 .
gateway.httpsurl		HTTPS 역 프록시 URL 을 입력하여 게이트웨이에서 페이지를 다시 쓸 사용자 정의 URL 을 설정합니다 .

표 2-1 platform.conf 파일 속성

항목	기본값	설명
gateway.favicon		게이트웨이에서 favicon.ico 파일에 대한 요청을 리디렉션할 URL 을 지정합니다 . 이는 Internet Explore 및 Netscape 7.0 이상의 기본 설정이나 즐겨찾기에 있는 "favorite icon" 에 사용됩니다 . 이 필드가 비어 있으면 게이트웨이는 “404 찾을 수 없습니다” 라는 메시지를 브라우저로 반환합니다 .
gateway.logging.password		이 필드에는 게이트웨이에서 응용프로그램 세션을 만드는 데 사용하는 사용자 "amService-srapGateway" 의 LDAP 비밀번호가 들어 있습니다 . 암호화되었거나 일반 텍스트일 수 있습니다 .
http.proxyHost		이 프록시 호스트는 Portal Server 에 접촉할 때 사용됩니다 .
http.proxyPort		이것은 Portal Server 에 접촉할 때 사용되는 호스트의 포트입니다 .
http.proxySet		이 속성은 프록시 호스트가 필요한 경우에 true로 설정됩니다 . 이 속성이 false 로 설정되면 http.proxyHost 및 http.proxyPort 가 무시됩니다 .
portal.server.*instance*		이 속성의 값은 해당 /etc/opt/SUNWam/config/AMConfig-*instance-name*.properties 파일입니다 . 기본값인 경우에는 AMConfig-default.properties 를 가리킵니다 .
gateway.cdm.cacheSleepTime		캐시 클라이언트 검색 모듈의 응답을 Identity Server 에서 게이트웨이로 보내는 경우의 시간 제한 값입니다 .
gateway.cdm.cacheCleanupTime		캐시 클라이언트 검색 모듈의 응답을 Identity Server 에서 게이트웨이로 보내는 경우의 시간 제한 값입니다 .

chroot 환경에서 게이트웨이 실행

chroot 환경에서 보안을 높이려면 chroot 디렉토리 컨텐트가 가능한 적어야 합니다. 예를 들어 , 사용자가 chroot 디렉토리의 파일을 수정할 수 있는 프로그램이 있으면 chroot 는 chroot 트리에서 파일을 수정하는 공격자로부터 서버를 보호하지 않습니다 . CGI 프로그램은 bourne shell, c-shell, korn shell 또는 perl 과 같은 해석된 언어로 작성되어서는 안되며 해석자가 chroot 디렉토리 트리에 놓지 않아도 되도록 이진수로 컴파일되어야 합니다 .

참고 워치독 기능은 chroot 환경에서는 지원되지 않습니다 .

➤ **chroot 를 설치하려면**

1. 루트로서 단말기 창에서 다음 파일을 네트워크에 있는 컴퓨터나 백업 테이프 또는 플로피 디스크와 같은 외부 소스로 복사합니다 .

   `cp /etc/vfstab` *external-device*

   `cp /etc/nsswitch.conf` *external-device*

   `cp /etc/hosts` *external-device*

2. 다음 디렉토리에서 `mkchroot` 스크립트를 실행합니다 .

   portal-server-install-root`/SUNWps/bin/chroot`

참고 실행되기 시작하면 `mkchroot` 스크립트는 Ctrl-C 를 눌러 종료할 수 없습니다 .

`mkchroot` 스크립트를 실행하는 동안 오류가 발생하면 49 페이지의 "mkchroot 스크립트의 실행 실패 " 를 참조하십시오 .

다른 루트 디렉토리를 입력하라는 메시지가 나타납니다 (new_root_directory). 스크립트에서 새 디렉토리를 만듭니다 .

다음 예제에서는 `/safedir/chroot` 가 new_root_directory 입니다 .

```
mkchroot version 6.0

Enter the full path name of the directory which will be the chrooted
tree:/safedir/chroot
Using /safedir/chroot as root.
Checking available disk space...done
```

```
mkchroot version 6.0
/safedir/chroot is on a setuid mounted partition.
Creating filesystem structure...dev etc sbin usr var proc opt bin lib tmp
etc/lib usr/platform usr/bin usr/sbin usr/lib usr/openwin/lib var/opt
var/tmp dev/fd done
Creating devices...null tcp ticots ticlts ticotsord tty udp zero conslog
done
Copying/creating etc files...group passwd shadow hosts resolv.conf netconfig
nsswitch.conf
done
Copying binaries.................................done
Copying libraries................................done
Copying zoneinfo (about 1 MB)..done
Copying locale info (about 5 MB).........done
Adding comments to /etc/nsswitch.conf ...done
Creating loopback mount for/safedir/chroot/usr/java1.2...done
Creating loopback mount for/safedir/chroot/proc...done
Creating loopback mount for/safedir/chroot/dev/random...done
Do you need /dev/fd (if you do not know what it means, press return)[n]:
Updating /etc/vfstab...done
Creating a /safedir/chroot/etc/mnttab file, based on these loopback mounts.
Copying SRAP related data ...
Using /safedir/chroot as root.
Creating filesystem structure..........done
mkchroot successfully done.
```

3. `platform.conf` 파일에 언급된 Java 디렉토리를 다음 명령을 사용하여 수동으로 chroot 디렉토리에 마운트합니다 .

```
mkdir -p /safedir/chroot/java-dir
```

```
mount -F lofs java-dir /safedir/chroot/java-dir
```

Solaris 9 에는 다음을 수행합니다 .

```
mkdir -p /safedir/chroot/usr/lib/32
```

```
mount -F lofs /usr/lib/32 /safedir/chroot/usr/lib/32
```

```
mkdir -p /safedir/chroot/usr/lib/64
```

```
mount -F lofs /usr/lib/64 /safedir/chroot/usr/lib/64
```

시스템을 시작할 때 이 디렉토리를 탑재하려면 `/etc/vfstab` 파일에 해당 항목을 추가합니다 .

java-dir - /safedir/chroot/*java-dir* lofs - no -

Solaris 9 의 경우 :

```
/usr/lib/32 - /safedir/chroot/usr/lib/32 lofs - no -
/usr/lib/64 - /safedir/chroot/usr/lib/64 lofs - no -
```

4. 아래 명령을 입력하여 게이트웨이를 다시 시작합니다 .

```
chroot /safedir/chroot ./gateway-install-root/SUNWps/bin/gateway start
stopping gateway ... done.
starting gateway ...
done.
```

mkchroot 스크립트의 실행 실패

mkchroot 스크립트를 실행하는 동안 오류가 발생하면 스크립트는 파일을 초기 상태로 복원합니다 .

다음 예제에서는 /safedir/chroot 가 chroot 디렉토리입니다 .

다음 오류 메시지가 발생한 경우 ,

```
Not a Clean Exit
```

1. 절차 chroot 를 설치하려면의 1 단계에서 백업 파일을 원래 위치로 복사하고 다음 명령을 실행합니다 .

```
umount /safedir/chroot/usr/java1.2
umount /safedir/chroot/proc
umount /safedir/chroot/dev/random
```

2. /safedir/chroot 디렉토리를 제거합니다 .

chroot 환경에서 게이트웨이 다시 시작

게이트웨이 시스템을 재부트할 때마다 chroot 환경에서 게이트웨이를 시작하려면 다음 단계를 수행합니다 .

➤ **chroot 환경에서 게이트웨이를 다시 시작하려면**

1. '/' 디렉토리에서 실행 중인 게이트웨이를 중지합니다 .

```
gateway-install-root/SUNWps/bin/gateway -n gateway-profile-name stop
```

2. chroot 디렉토리에서 실행할 게이트웨이를 시작합니다 .

   `chroot /safedir/chroot ./`*`portal-server-install-root`*`/SUNWps/bin/gateway -n `*`gateway-profile-name`*` start`

참고 `/safedir/chroot/etc` 파일 (`passwd` 및 `hosts` 등) 은 `/etc` 파일과 같이 관리가 필요하지만 chroot 트리에서 실행되는 프로그램에 필요한 호스트 및 계정 정보만 들어 있습니다 .

예를 들어 , 시스템의 identity 공급자 주소를 변경하는 경우에는 파일 `/safedir/chroot/etc/hosts` 도 변경합니다 .

게이트웨이의 다중 인스턴스 만들기

`gwmultiinstance` 스크립트를 사용하여 게이트웨이의 새 인스턴스를 만듭니다 . 게이트웨이 프로필을 만든 후에 이 스크립트를 실행하는 것이 좋습니다 .

1. 루트로 로그인하여 다음 디렉토리로 찾아 갑니다 .

   `gateway-install-root``/SUNWps/bin/`

2. 다중 인스턴스 스크립트를 실행합니다 .

   `./gwmultiinstance`

3. 다음 설치 옵션 중 하나를 선택합니다 .

```
1) Create a new gateway instance
2) Remove a gateway instance
3) Remove all gateway instances
4) Exit
```

1 을 선택한 경우 다음 질문에 답하십시오 .

```
What is the name of the new gateway instance?
What protocol will the new gateway instance use? [https]
What port will the new gateway instance listen on?
What is the fully qualified hostname of the portal server?
What port should be used to access the portal server?
What protocol should be used to access the portal server? [http]
What is the portal server deploy URI?
What is the organization DN? [dc=iportal,dc=com]
What is the identity server URI? [/amserver]
What is the identity server password encryption key?

Please provide the following information needed for creating a
self-signed certificate:
What is the name of your organization?
What is the name of your division?
What is the name of your city or locality?
What is the name of your state or province?
What is the two-letter country code?
What is the password for the Certificate Database? Again?

What is the password for the logging user? Again?
Have you created the new gateway profile in the admin console? [y]/n
Start the gateway after installation? [y]/n
```

4. 새 게이트웨이 프로필 이름으로 게이트웨이의 새 인스턴스를 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

   여기서 *gateway-profile-name* 은 새 게이트웨이 인스턴스입니다 .

게이트웨이 프로필 외에도 AMConfig-.*instance-name*.properties 파일이 /etc/opt/SUNWam/config 디렉토리에 만들어집니다 .

platform.conf 파일에 portal.server.instance 속성이 있으면 게이트웨이에서 그에 해당하는 AMConfig-*instance-name*.properties 파일을 읽습니다. platform.conf 파일에 portal.server.instance 속성이 없으면 게이트웨이에서 기본 AMConfig 파일 (AMConfig.properties) 을 읽습니다 .

홈이 여러 개인 게이트웨이 인스턴스 만들기

홈이 여러 개인 게이트웨이 인스턴스를 만드는 경우 , 즉 한 Portal Server 에 여러 게이트웨이를 만드는 경우에는 다음과 같이 platform.conf 파일을 수정해야 합니다 .

```
gatewaybindipaddress = 0.0.0.0
```

같은 LDAP 를 사용하여 게이트웨이 인스턴스 만들기

같은 LDAP 를 사용하는 게이트웨이 인스턴스 여러 개를 만드는 경우에는 첫 게이트웨이를 만든 후에 그 뒤의 모든 게이트웨이에서 다음을 수행합니다 .

/etc/opt/SUNWam/config/ 에서 AMConfig-*instance-name*.properties 의 다음 영역을 처음 설치한 게이트웨이 인스턴스와 일치하도록 수정합니다 .

1. 암호의 암호화와 해독에 사용되는 키를 첫 게이트웨이와 같은 문자열로 대체합니다 .

   am.encryption.pwd= *string_key_specified_in gateway-install*

2. 응용 프로그램 인증 모듈의 공유 비밀에 해당하는 키를 대체합니다 .

   com.iplanet.am.service.secret= *string_key_specified_in gateway-install*

3. /etc/opt/SUNWam/config/ums 에서 serverconfig.xml 의 다음 영역을 처음 설치한 Portal-Identity Server 와 다른 값으로 수정합니다 .

   <DirDN> cn=puser,ou=DSAME Users,dc=sun,dc=net</DirDN>

   <DirPassword>*string_key_specified_in gateway-install*</DirPassword>

   <DirDN>cn=dsameuser,ou=DSAME Users,dc=sun,dc=net</DirDN>

   <DirPassword>*string_key_specified_in gateway-install* </DirPassword>

4. amserver 서비스를 다시 시작합니다 .

게이트웨이 시작 및 중지

기본적으로 게이트웨이는 사용자 noaccess 로 시작됩니다 .

➤ **게이트웨이를 시작하려면**

1. 게이트웨이를 설치하고 필요한 프로필을 만든 후에는 다음 명령을 실행하여 게이트웨이를 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n default start

   default 는 설치 중에 만들어지는 기본 게이트웨이 프로필입니다 . 나중에 고유한 프로필을 만들고 새 프로필로 게이트웨이를 다시 시작할 수 있습니다 . 38 페이지의 " 게이트웨이 프로필 만들기 " 를 참조하십시오 .

   다중 게이트웨이 인스턴스가 있다면 다음을 사용합니다 .

   gateway-install-root/SUNWps/bin/gateway start

이 명령은 특정 컴퓨터에 구성된 모든 게이트웨이 인스턴스를 시작합니다 .

참고 서버를 다시 시작하면 (게이트웨이의 인스턴스를 구성한 서버) 게이트웨이의 구성된 인스턴스가 모두 다시 시작됩니다 .

/etc/opt/SUNWps 디렉토리에 기존 프로필이나 백업 프로필이 없어야 합니다 .

2. 다음 명령을 실행하여 지정 포트에서 게이트웨이가 실행되고 있는지 점검합니다 .

   netstat -a | grep *port-number*

   기본 게이트웨이 포트는 443 입니다 .

➤ **게이트웨이를 중지하려면**

1. 게이트웨이를 중지하려면 다음 명령을 사용합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* stop

   다중 게이트웨이 인스턴스가 있으면 다음을 사용합니다 .

   gateway-install-root/SUNWps/bin/gateway stop

   이 명령은 특정 컴퓨터에서 실행되고 있는 모든 게이트웨이 인스턴스를 중지합니다 .

2. 다음 명령을 실행하여 게이트웨이 프로세스가 더 이상 실행되지 않는지 확인합니다 .

   /usr/bin/ps -ef | entsys

게이트웨이 다시 시작

일반적으로는 게이트웨이를 다시 시작할 필요가 없습니다 . 다음 이벤트가 발생한 경우에만 다시 시작합니다 .

- 새 프로필을 만들었으며 이 새 프로필을 게이트웨이에 할당해야 하는 경우 .
- 기존 프로필의 일부 속성을 수정하였으며 변경 사항을 적용해야 하는 경우 .

➤ **다른 프로필로 게이트웨이를 다시 시작하려면**

게이트웨이를 다시 시작합니다 .

gateway-install-root/SUNWps/bin/gateway -n *new-gateway-profile-name* start

➤ **게이트웨이를 다시 시작하려면**

단말기 창에서 루트로 연결하고 다음 작업 중 하나를 수행합니다 .

- 워치독 프로세스를 시작합니다 .

gateway-install-root/SUNWps/bin/gateway watchdog on

그러면 crontab 에 항목이 만들어지고 워치독 프로세스가 활성 상태가 됩니다 . 워치독은 특정 컴퓨터 및 게이트웨이 포트에서 실행 중인 모든 게이트웨이 인스턴스를 모니터링하여 다운된 경우 게이트웨이를 다시 시작합니다 .

➤ **게이트웨이 워치독을 구성하려면**

워치독이 게이트웨이의 상태를 모니터링하게 될 시간 간격을 설정할 수 있습니다 . 시간 간격은 기본적으로 60 초로 설정됩니다 . 이 기본 설정을 변경하려면 crontab 에서 다음 라인을 편집합니다 .

0-59 * * * * *gateway-install-root*/SUNWps/bin/

/var/opt/SUNWps/.gw. 5> /dev/null 2>&1

crontab 항목을 구성하려면 crontab 맨 페이지를 참조하십시오 .

가상 호스트 지정

가상 호스트는 같은 시스템 IP 와 호스트 이름을 가리키는 추가 호스트 이름입니다 . 예를 들어 호스트 이름 a.b.c 가 호스트 IP 주소 192.155.205.133 을 가리키는 경우 , 같은 IP 주소를 가리키는 다른 호스트 이름 c.d.e 를 추가할 수 있습니다 .

➤ **가상 호스트를 지정하려면**

1. 루트로 로그인하여 필요한 게이트웨이 인스턴스의 platform.conf 파일을 편집합니다 .

   /etc/opt/SUNWps/platform.conf.*gateway-profile-name*

2. 다음 항목을 추가합니다 .

   gateway.virtualhost=*fully-qualified-gateway-host gateway-ip-address fully-qualified-reverse-proxyhost*

   gateway.enable.customurl=true (이 값은 기본적으로 false 로 설정됩니다 .)

3. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

이 값이 지정되어 있지 않으면 게이트웨이는 기본적으로 일반 작동을 사용합니다 .

Identity Server 에 접속하도록 프록시 지정

게이트웨이에서 호스트 프록시를 사용하여 Portal Server 에 배포되는 SRA 코어 (RemoteConfigServlet) 에 접속하도록 지정할 수 있습니다 . 이 프록시는 게이트웨이가 Portal Server 와 Identity Server 에 접속하기 위해 사용됩니다 .

➤ **프록시를 지정하려면**

1. 명령줄에서 다음 파일을 편집합니다 .

   `/etc/opt/SUNWps/platform.conf.`*gateway-profile-name*

2. 다음 항목을 추가합니다 .

   `http.proxyHost=`*proxy-host*

   `http.proxyPort=`*proxy-port*

   `http.proxySet=true`

3. 서버에 대한 요청에 지정된 프록시를 사용할 수 있도록 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

웹 프록시 사용

타사 웹 프록시를 사용하여 HTTP 리소스에 연결하도록 게이트웨이를 구성할 수 있습니다 . 웹 프록시는 클라이언트와 인터넷 사이에 상주합니다 .

웹 프록시 구성

여러 도메인 및 부속 도메인에 서로 다른 프록시가 사용될 수 있습니다 . 이 항목은 특정 도메인에서 특정 부속 도메인에 연결할 때 어떤 프록시를 사용할지 게이트웨이에 알려 줍니다 . 게이트웨이에 지정된 프록시 구성은 다음과 같이 작동합니다 .

- 게이트웨이 서비스의 [도메인 및 부속 도메인의 프록시] 필드에 필요한 프록시와 함께 도메인 및 부속 도메인 목록을 만듭니다 .

  도메인 및 부속 도메인의 프록시 구성에 대한 자세한 내용은 260 페이지의 " 도메인 및 부속 도메인의 프록시 목록 만들기 " 를 참조하십시오 .

- 프록시 사용 옵션을 사용 설정해 놓은 경우 .

  - [도메인 및 부속 도메인의 프록시] 필드에 지정된 프록시가 지정된 호스트에 사용됩니다 .

- 도메인 및 부속 도메인의 프록시 목록에 지정된 도메인 및 부속 도메인에서 특정 URL 에 직접 연결하려면 [웹 프록시 URL 사용 안함] 필드에서 해당 URL 을 지정합니다 .

- 프록시 사용 옵션을 사용 해제해 놓은 경우 .

  - 프록시가 도메인 및 부속 도메인의 프록시 필드에 지정된 도메인과 부속 도메인에서 특정 URL 에 사용되도록 하려면 [웹 프록시 URL 사용] 목록에서 해당 URL 을 지정합니다 . 프록시 사용 옵션이 사용 해제되어 있더라도 [웹 프록시 사용] 에 나열된 URL 에 프록시를 사용하여 연결할 수 있습니다 . 이 URL 의 프록시는 [도메인 및 부속 도메인의 프록시] 목록에서 가져온 것입니다 .

프록시 사용 옵션을 구성하려면 258 페이지의 " 웹 프록시 사용 활성화 " 를 참조하십시오 .

그림 2-1 은 게이트웨이 서비스의 프록시 구성에 기반하여 웹 프록시 정보가 어떻게 결정되는지 보여줍니다 .

그림 2-1 웹 프록시 관리

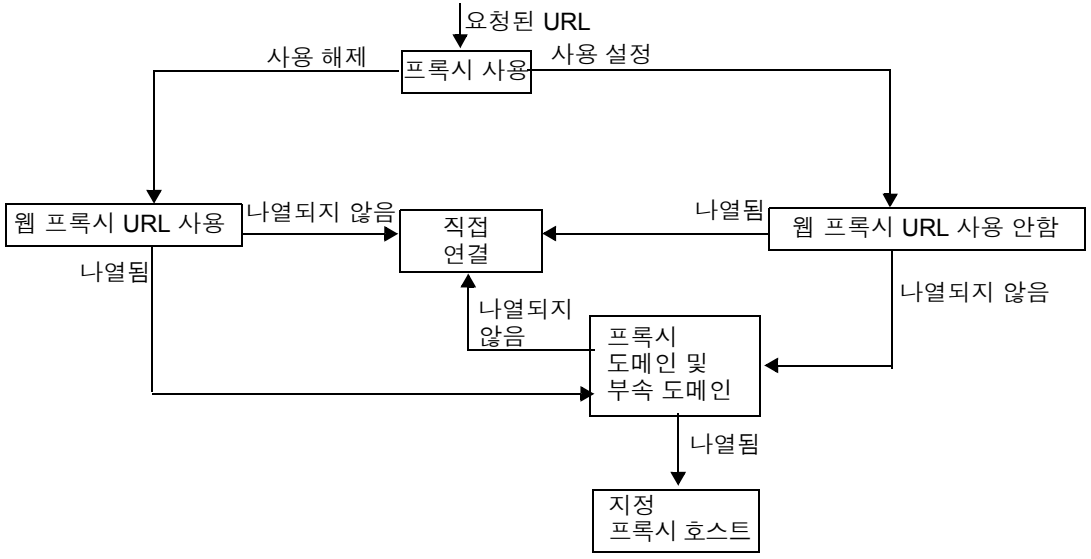


그림 2-1 에서 프록시 사용을 사용 설정해 놓았고 , 요청된 URL 이 [웹 프록시 URL 사용 안함] 목록에 나열되는 경우 게이트웨이가 대상 호스트에 직접 연결됩니다 .

프록시 사용을 사용 설정해 놓았고 , 요청된 URL 이 [웹 프록시 URL 사용 안함] 목록에 나열되는 않는 경우 게이트웨이는 지정된 프록시를 통해 대상 호스트에 연결됩니다 . 프록시가 지정되어 있는 경우에는 [도메인 및 부속 도메인의 프록시] 목록에서 찾으면 됩니다 .

프록시 사용을 사용 해제하였고 , 요청된 URL 이 [웹 프록시 URL 사용] 목록에 나열되면 게이트웨이는 [도메인 및 부속 도메인의 프록시] 목록에 있는 프록시 정보를 사용하여 대상 호스트에 연결됩니다 .

프록시 사용을 사용 해제하였고 , 요청된 URL 이 [웹 프록시 URL 사용] 목록에 나열되지 않으면 게이트웨이가 대상 호스트에 직접 연결됩니다 .

위에 설명된 조건 중 어느 것에도 해당하지 않아서 직접 연결이 불가능하면 게이트웨이는 연결할 수 없다는 오류 메시지를 표시합니다 .

참고 표준 포털 데스크탑의 책갈피 채널을 통해 URL 에 액세스하는 중에 위에 설명된 조건 중 어느 것도 충족되지 않으면 게이트웨이는 브라우저로 리디렉션합니다 . 그러면 브라우저는 자체 프록시 설정을 통해 URL 에 액세스합니다 .

구문

```
domainname [web_proxy1:port1]|subdomain1 [web_proxy2:port2]|......
```

예

```
sesta.com wp1:8080|red wp2:8080|yellow|* wp3:8080
```

* 는 모든 항목과 일치되는 와일드카드입니다 .

여기서 ,

sesta.com 은 도메인 이름이고 wp1 은 포트 8080 에 연결할 프록시입니다 .

red 는 부속 도메인이고 wp2 는 포트 8080 에 연결할 프록시입니다 .

yellow 는 부속 도메인입니다 . 프록시가 지정되어 있지 않고 포트 8080 에 도메인에 지정된 프록시 즉 , wp1 이 사용됩니다 .

* 는 모든 다른 부속 도메인에서 포트 8080 에 wp3 을 사용해야 함을 나타냅니다 .

참고 포트를 지정하지 않은 경우 기본적으로 포트 8080 이 사용됩니다 .

웹 프록시 정보 처리

클라이언트에서 특정 URL 에 액세스하려고 할 때 URL 의 호스트 이름은 [도메인 및 부속 도메인의 프록시] 목록에 있는 항목과 일치합니다 . 요청된 호스트 이름의 가장 긴 접미어에 일치하는 항목이 선택됩니다 . 예를 들어 , 요청된 호스트 이름이 host1.sesta.com 이라고 고려해 봅시다 .

- [도메인 및 부속 도메인의 프록시] 에 host1.sesta.com 이 없는지 스캔됩니다 . 일치하는 항목이 있으면 이 항목에 지정된 프록시를 통해 그 호스트에 연결됩니다 .
- 그렇지 않으면 목록에 *.sesta.com 이 없는지 스캔됩니다 . 항목을 찾으면 해당 프록시가 사용됩니다 .
- 그렇지 않으면 목록에 sesta.com 이 없는지 검색됩니다 . 항목을 찾으면 해당 프록시가 사용됩니다 .
- 그렇지 않으면 목록에 *.com 이 없는지 검색됩니다 . 항목을 찾으면 해당 프록시가 사용됩니다 .
- 그렇지 않으면 목록에 com 이 없는지 검색됩니다 . 항목을 찾으면 해당 프록시가 사용됩니다 .
- 그렇지 않으면 목록에 * 이 없는지 검색됩니다 . 항목을 찾으면 해당 프록시가 사용됩니다 .
- 그렇지 않으면 직접 연결이 시도됩니다 .

[도메인 및 부속 도메인의 프록시] 목록에서 다음 항목을 고려합니다 .

```
com p1| host1 p2 | host2 | * p3

sesta.com p4 | host5 p5 | * p6

florizon.com | host6

abc.sesta.com p8 | host7 p7 | host8 p8 | * p9

host6.florizon.com p10

host9.sesta.com p11

siroe.com | host12 p12 | host13 p13 | host14 | * p14

siroe.com | host15 p15 | host16 | * p16

* p17
```

게이트웨이는 표 2-2 에 나와 있듯이 이 항목을 테이블에 내부적으로 매핑합니다 .

표 2-2 [도메인 및 부속 도메인의 프록시] 목록에서 항목 매핑

번호	도메인 및 부속 도메인의 프록시 목록의 항목	프록시	설명
1	com	p1	목록에 지정된 대로 .
2	host1.com	p2	목록에 지정된 대로 .
3	host2.com	p1	host2 에는 프록시가 지정되어 있지 않기 때문에 도메인의 프록시가 사용됩니다 .
4	*.com	p3	목록에 지정된 대로 .
5	sesta.com	p4	목록에 지정된 대로 .
6	host5.sesta.com	p5	목록에 지정된 대로 .
7	*.sesta.com	p6	목록에 지정된 대로 .
8	florizon.com	직접	자세한 내용은 항목 14 에 대한 설명 참조 .
9	host6.florizon.com	–	자세한 내용은 항목 14 에 대한 설명 참조 .
10	abc.sesta.com	p8	목록에 지정된 대로 .
11	host7.abc.sesta.com	p7	목록에 지정된 대로 .
12	host8.abc.sesta.com	p8	목록에 지정된 대로 .
13	*.abc.sesta.com	p9	목록에 지정된 대로 . abc.sesta.com 도메인에서 host7 과 host8 을 제외한 모든 호스트에는 p9 가 프록시로 사용됩니다 .
14	host6.florizon.com	p10	이 항목은 항목 9 와 동일합니다 . 그러나 항목 9 는 직접 연결을 나타내지만 , 이 항목은 프록시 p10 을 사용해야 함을 나타냅니다 . 이 경우와 같이 2 개 항목이 있는 경우에는 프록시 정보가 있는 항목이 유효한 항목으로 간주됩니다 . 다른 항목은 무시됩니다 .
15	host9.sesta.com	p11	목록에 지정된 대로 .
16	siroe.com	직접	siroe.com 에는 지정된 프록시가 없기 때문에 직접 연결이 시도됩니다 .
17	host12.siroe.com	p12	목록에 지정된 대로 .
18	host13.siroe.com	p13	목록에 지정된 대로 .
19	host14.siroe.com	직접	host14 또는 siroe.com 에는 지정된 프록시가 없기 때문에 직접 연결이 시도됩니다 .

표 2-2 [도메인 및 부속 도메인의 프록시] 목록에서 항목 매핑

번호	도메인 및 부속 도메인의 프록시 목록의 항목	프록시	설명
20	*.siroe.com	p14	항목 23 에 대한 설명 참조 .
21	host15.siroe.com	p15	목록에 지정된 대로 .
22	host16.siroe.com	직접	host16 또는 siroe.com 에는 지정된 프록시가 없기 때문에 직접 연결이 시도됩니다 .
23	*.siroe.com	p16	이 항목은 항목 20 과 비슷하지만 지정된 프록시가 다릅니다 . 이런 경우 게이트웨이의 정확한 동작은 알 수 없습니다 . 두 프록시 중 하나가 사용됩니다 .
24	*	p17	요청된 URL 과 일치하는 다른 항목이 없으면 p17 이 프록시로 사용됩니다 .

참고 [도메인 및 부속 도메인의 프록시] 목록에서 프록시 항목을 | 기호와 분리하는 것보다 목록에 개별 항목을 보유하는 것이 더 간단할 수 있습니다 . 예를 들어 , 다음과 같은 항목 대신에

```
sesta.com p1 | red p2 | * p3
```

이 항목을 다음과 같이 지정할 수 있습니다 .

```
sesta.com p1

red.sesta.com p2

*.sesta.com p3
```

그러면 쉽게 반복되는 항목이나 기타 모호함의 범위를 좁힐 수 있습니다 .

도메인 및 부속 도메인의 프록시 목록에 기반하여 다시 쓰기

[도메인 및 부속 도메인의 프록시] 목록의 항목도 Rewriter 에서 사용됩니다 . Rewriter 는 도메인이 [도메인 및 부속 도메인의 프록시] 목록에 나열된 도메인과 일치하는 모든 URL 을 다시 씁니다 .

주의 [도메인 및 부속 도메인의 프록시] 목록의 * 항목은 다시 쓰기에 고려되지 않습니다 . 예를 들어 , 표 2-2 에 나온 예제에서는 항목 24 가 고려되지 않습니다 .

Rewriter 에 대한 자세한 내용은 3 장 "Proxylet 및 Rewriter" 를 참조하십시오 .

기본 도메인 및 부속 도메인

URL 의 대상 호스트가 완전한 정규 호스트 이름이 아닐 경우 , 완전한 정규 이름에 도달하도록 기본 도메인 및 부속 도메인을 사용합니다 .

관리 콘솔의 [기본 도메인] 필드 항목이 다음과 같다고 가정해 봅시다 .

```
red.sesta.com
```

참고 [도메인 및 부속 도메인의 프록시] 목록에 상응하는 항목이 있어야 합니다 .

위의 예에서는 `sesta.com` 이 기본 도메인이고 기본 부속 도메인은 `red` 입니다 .

요청된 URL 이 `host1` 인 경우 , 기본 도메인 및 부속 도메인을 통해 `host1.red.sesta.com` 으로 결정됩니다 . 그리고 나서 [도메인 및 부속 도메인의 프록시] 목록에 `host1.red.sesta.com` 이 없는지 검색됩니다 .

자동 프록시 구성 사용

[도메인 및 부속 도메인의 프록시] 목록에 있는 정보를 무시하려면 자동 프록시 구성 (PAC) 기능을 활성화합니다 . 이를 구성하려면 261 페이지의 " 자동 프록시 구성 지원 사용 " 을 참조하십시오 .

자동 프록시 구성 (PAC) 파일을 사용할 때는 다음을 주의합니다 .

- `js.jar` 는 게이트웨이 컴퓨터의 `$JRE_HOME/lib/ext` 디렉토리에 있어야 합니다 . 그렇지 않으면 게이트웨이가 PAC 파일의 구문을 분석할 수 없습니다 .
- 게이트웨이는 부팅 시에 게이트웨이 프로필 [자동 프록시 구성 파일] 위치 필드에 지정된 위치로부터 PAC 파일을 불러옵니다 . 위치를 구성하려면 262 페이지의 " 자동 프록시 구성 파일 위치 지정 " 을 참조하십시오 .
- Portal Server, 게이트웨이 , Netlet, Proxylet, Jchardet 는 Rhino 소프트웨어를 사용하여 PAC 파일을 해석합니다 . 이 소프트웨어는 SUNWrhino 패키지에서 설치할 수 있습니다 .

이 패키지에는 /usr/share/lib 디렉토리에 꼭 필요한 js.jar 파일이 포함되어 있습니다. 이 디렉토리를 게이트웨이 및 Portal Server 시스템의 webserver/appserver 클래스 경로에 추가합니다. 그렇게 하지 않으면 Portal Server, 게이트웨이, Netlet, Proxylet, Jchardet 에서 PAC 파일을 구문 분석할 수 없습니다.

- 게이트웨이는 URLConnection API 를 사용하여 이 위치에 도달합니다. 프록시가 이 위치에 도달하도록 구성해야 하는 경우에는 프록시를 다음과 같이 구성해야 합니다.

  a. 명령줄에서 다음 파일을 편집합니다.

  /etc/opt/SUNWps/platform.conf.*gateway-profile-name*

  b. 다음 항목을 추가합니다.

  http.proxyHost=*web-proxy-hostname*

  http.proxyPort=*web-proxy-port*

  http.proxySet=true

  c. 게이트웨이를 다시 시작하여 지정된 프록시를 사용합니다.

  gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

- PAC 파일 초기화가 실패하면 게이트웨이는 [도메인 및 부속 도메인의 프록시] 목록에 있는 정보를 사용합니다.
- PAC 파일로부터 "" (빈 문자열) 이나 "null" 이 반환되면 게이트웨이에서는 호스트가 인트라넷에 속하지 않는다고 가정합니다. 이는 호스트가 [도메인 및 부속 도메인의 프록시] 목록에 있지 경우와 비슷합니다.

  게이트웨이에서 호스트에 직접 연결되도록 하려면 "DIRECT" 을 반환합니다. 64 페이지의 "DIRECT 또는 NULL 이 반환되는 예제 " 를 참조하십시오.
- 여러 프록시가 지정되어 있으면 게이트웨이는 첫 번째 반환된 프록시만 사용합니다. 호스트에 지정된 여러 프록시에서 페일오버나 로드 균형 조정을 시도하지 않습니다.
- 게이트웨이는 SOCKS 프록시를 무시하고 직접 연결을 시도하면서 호스트가 인트라넷의 일부라 가정합니다.
- 인트라넷의 일부가 아닌 호스트에 도달하는 데 프록시를 사용하도록 지정하려면 프록시 유형 "STARPROXY" 를 사용합니다. 이 유형은 PAC 파일 형식의 확장이며 게이트웨이 프로필에 있는 [도메인 및 부속 도메인의 프록시] 섹션의 항목 * proxyHost:port 와 유사합니다. 64 페이지의 "STARPROXY 가 반환되는 예제 " 를 참조하십시오.

예제 PAC 파일 사용

다음 예제는 [도메인 및 부속 도메인의 프록시] 목록과 상응하는 PAC 파일에 나열된 URL 을 보여줍니다 .

DIRECT 또는 NULL 이 반환되는 예제

도메인 및 부속 도메인에 이 프록시 사용 :

```
*intranet1.com proxy.intranet.com:8080

intranet2.com proxy.intranet1.com:8080
```

상응하는 PAC 파일 :

```
// Start of the PAC File
function FindProxyForURL(url, host) {
         if (dnsDomainIs(host, ".intranet1.com")) {
             return "DIRECT";
         }
          if (dnsDomainIs(host, ".intranet2.com")) {
              return "PROXY proxy.intranet1.com:8080";
          }
          return "NULL";
}
//End of the PAC File
```

STARPROXY 가 반환되는 예제

도메인 및 부속 도메인에 이 프록시 사용 :

```
intranet1.com

intranet2.com.proxy.intranet1.com:8080

internetproxy.intranet1.com:80
```

상응하는 PAC 파일 :

```
// Start of the PAC File
function FindProxyForURL(url, host) {
          if (dnsDomainIs(host, ".intranet1.com")) {
```

```
            return "DIRECT";
        }
        if (dnsDomainIs(host, ".intranet2.com")) {
            return "PROXY proxy.intranet1.com:8080;" +
                "PROXY proxy1.intranet1.com:8080";
        }
        return "STARPROXY internetproxy.intranet1.com:80";
}
//End of the PAC File
```

이 경우 요청이 .intranet2.com domain 에 있는 호스트에 대한 것이면 게이트웨이는 proxy.intranet1.com:8080 에 연결합니다. 프록시 intranet1.com:8080 이 다운되면 요청이 실패합니다. 그래도 게이트웨이에서는 장애 조치를 수행하고 proxy1.intranet1.com:8080 에 연결하지 않습니다.

Netlet 프록시 사용

Netlet 패킷은 게이트웨이에서 비밀번호가 해독되어 대상 서버로 보내집니다. 그러나 게이트웨이는 비무장 지대 (DMZ) 와 인트라넷 사이의 방화벽을 통해 모든 Netlet 대상 호스트에 액세스해야 합니다. 그러려면 방화벽에서 많은 포트를 열어야 합니다. Netlet 프록시는 방화벽에서 열린 포트의 수를 줄이는 데 사용할 수 있습니다.

Netlet 프록시는 클라이언트로부터 게이트웨이를 거쳐 인트라넷에 상주하는 Netlet 프록시에 이르기까지 보안 터널을 확장하여 게이트웨이와 인트라넷 사이의 보안을 강화합니다. 프록시가 있으면 Netlet 패킷은 프록시에서 해독된 후 대상으로 보내집니다.

Netlet 프록시가 유용한 이유는 다음과 같습니다.

- 보안 계층을 추가할 수 있음.
- 상당한 규모의 배치 환경에서 내부 방화벽을 통해 추가 IP 주소와 게이트웨이의 포트 사용을 최대한 줄일 수 있음.
- 게이트웨이와 Portal Server 간 개방 포트 수를 1 로 제한할 수 있음. 이 포트 수는 설치 시 구성 가능.

- 클라이언트와 게이트웨이 사이의 보안 채널을 그림 2-2 의 "Netlet 프록시가 구성되어 있는 경우 " 부분에 나와 있듯이 Portal Server 까지 확장할 수 있음 . Netlet 프록시는 데이터 암호화를 통해 보안을 강화한다는 이점이 있지만 시스템 자원을 더 많이 사용할 수 있습니다 . Netlet 프록시 설치에 대한 자세한 내용은 *Sun Java Enterprise System 설치 설명서*를 참조하십시오 .

가능한 작업 :

- Portal Server 노드나 별도 노드에 Netlet 프록시를 설치할 수 있습니다 .
- 다중 Netlet 프록시를 설치하고 관리 콘솔을 사용하여 단일 게이트웨이에 구성할 수 있습니다 . 이는 로드 균형 조정에 유용합니다 . 자세한 내용은 244 페이지의 "Netlet 프록시 목록 활성화 및 만들기 " 를 참조하십시오 .
- 단일 시스템에 Netlet 프록시의 다중 인스턴스를 구성할 수 있습니다 .
- 게이트웨이의 다중 인스턴스를 Netlet 프록시의 단일 설치에 지정할 수 있습니다 .
- 웹 프록시를 통해 Netlet 을 통과할 수 있습니다 . 이를 구성하려면 263 페이지의 " 웹 프록시를 통한 Netlet 터널링 활성화 " 를 참조하십시오 .

그림 2-2 에는 Netlet 프록시가 설치된 경우와 설치되지 않은 경우 , 게이트웨이와 Portal Server 를 구현하는 3 가지 구현 샘플이 나와 있습니다 . 구성 요소에는 클라이언트 , 방화벽 2 개 , 두 방화벽 사이에 상주하는 게이트웨이 , Portal Server 및 Netlet 대상 서버가 포함됩니다 .

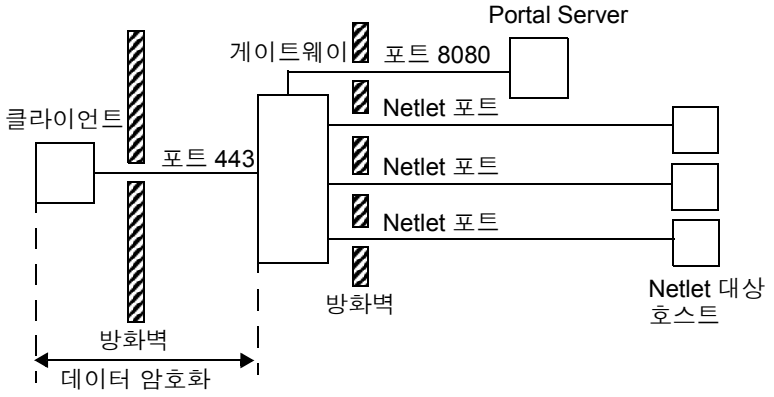
첫 번째 시나리오는 Netlet 프록시가 설치되지 않은 경우의 게이트웨이와 Portal Server 를 보여줍니다 . 여기서는 데이터 암호화가 클라이언트에서 게이트웨이까지만 적용됩니다 . 각 Netlet 연결 요청을 위해 두 번째 방화벽에서 포트가 1 개 개방되어 있습니다 .

두 번째 시나리오는 Netlet 프록시가 Portal Server 에 설치된 경우의 게이트웨이와 Portal Server 를 보여줍니다 . 이 경우 데이터 암호화는 클라이언트에서 Portal Server 까지 전체적으로 적용됩니다 . 모든 Netlet 연결이 Netlet 프록시를 통해 라우팅되기 때문에 두 번째 방화벽에서 Netlet 요청에 사용되는 포트는 하나만 열려 있으면 됩니다 .

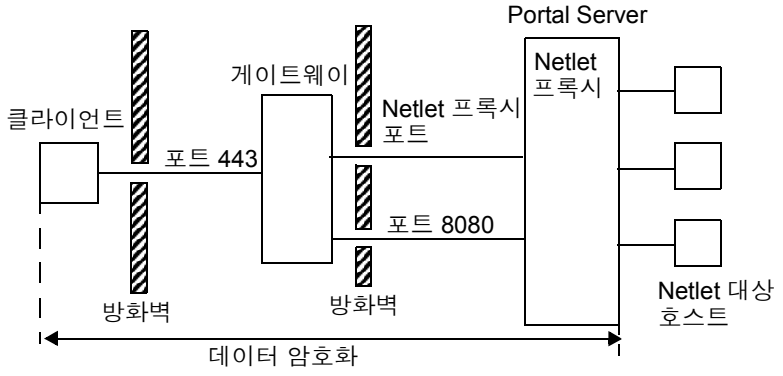
세 번째 시나리오는 Netlet 프록시가 별도 노드에 설치된 경우의 게이트웨이와 Portal Server 를 보여줍니다 . Netlet 프록시를 별도 노드에 설치하면 Portal Server 노드의 로드가 줄어듭니다 . 여기서는 두 번째 방화벽에서 2 개의 포트만 개방되어 있으면 됩니다 . 한 포트는 Portal Server 에 대한 요청을 처리하고 다른 포트는 Netlet 프록시 서버에 대한 Netlet 요청을 라우팅합니다 .

그림 2-2 Netlet 프록시 구현

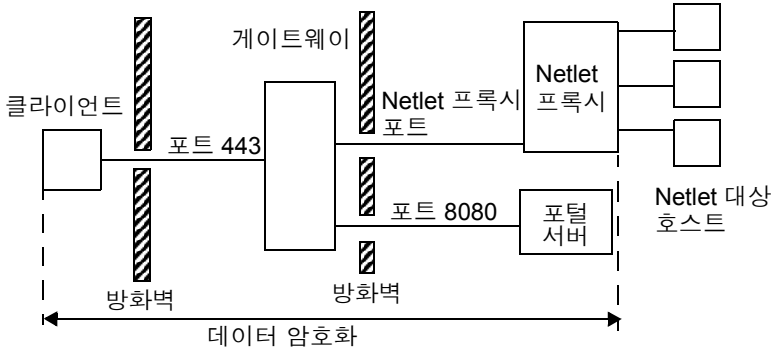
Netlet 프록시가 구성되지 않은 경우



Portal Server 에 Netlet 프록시가 있는 경우



별도 노드에 Netlet 프록시가 있는 경우



Netlet 프록시의 인스턴스 만들기

Portal Server 노드나 별도 노드에 Netlet 프록시의 새 인스턴스를 만들려면 `nlpmultiinstance` 스크립트를 사용합니다 . 게이트웨이 프로필을 만든 후에 이 스크립트를 실행하는 것이 좋습니다 .

1. 루트로 로그인하여 다음 디렉토리로 찾아 갑니다 .

   netlet-install-dir`/SUNWps/bin`

2. 다중 인스턴스 스크립트를 실행합니다 .

   `./nlpmultiinstance`

3. `nlpmultiinstance` 스크립트에서 나타나는 다음 질문에 답합니다 .
   - 새 netlet 프록시 인스턴스의 이름은 무엇입니까 ?
   - 이 노드에 같은 이름으로 구성된 인스턴스가 있으면 이 netlet 프록시 인스턴스에도 같은 구성을 사용할 것인지 묻는 메시지가 나타납니다 .
   - 예라고 답한 경우 , 다음 두 질문에 답하십시오 .
     - 새 netlet 프록시 인스턴스에서는 어떤 포트를 수신합니까 ?
     - 설치 후 netlet 프록시를 시작하시겠습니까 ?
   - 아니오라고 답한 경우 , 다음 질문에 답하십시오 .
     - 새 netlet 프록시 인스턴스에서는 어떤 프로토콜을 사용합니까 ?
     - 새 netlet 프록시 인스턴스에서는 어떤 포트를 수신합니까 ?
     - 조직의 이름은 무엇입니까 ?
     - 부서 이름은 무엇입니까 ?
     - 구 / 군 / 시의 이름은 무엇입니까 ?
     - 시 / 도의 이름은 무엇입니까 ?
     - 2 자로 된 국가 번호는 무엇입니까 ?
     - 인증서 데이터베이스 비밀번호는 무엇입니까 ?
     - 사용자 로그인 비밀번호는 무엇입니까 ?
     - 관리 콘솔에 새 게이트웨이 프로필을 만들었습니까 ?
     - 예라고 답한 경우 , 설치 후 netlet 프록시를 시작하시겠습니까 ?

4. 새 게이트웨이 프로필 이름으로 netlet 프록시의 새 인스턴스를 시작합니다 .

   netlet-proxy-install-root/SUNWps/bin/netletd -n *gateway-profile-name* start

   여기서 *gateway-profile-name* 은 필요한 게이트웨이 인스턴스에 해당하는 프로필 이름입니다 .

Netlet 프록시 사용 설정

Identity Server 관리 콘솔의 SRA 구성에서 게이트웨이 서비스를 통해 Netlet 프록시를 활성화합니다 . 244 페이지의 "Netlet 프록시 목록 활성화 및 만들기 " 를 참조하십시오 .

Netlet 프록시 다시 시작

프록시가 예기치 않게 중단될 때마다 다시 시작하도록 Netlet 프록시를 구성할 수 있습니다 . 워치독 프로세스를 예약하여 Netlet 프록시를 모니터하고 , 프록시가 다운된 경우 다시 시작할 수 있습니다 .

Netlet 프록시를 수동으로 다시 시작할 수도 있습니다 .

➤ Netlet 프록시를 다시 시작하려면

단말기 창에서 루트로 연결하고 다음 작업 중 하나를 수행합니다 .

- 워치독 프로세스를 시작합니다 .

  netlet-proxy-install-root/SUNWps/bin/netletd watchdog on

  그러면 crontab 에 항목이 만들어지고 워치독 프로세스가 활성 상태가 됩니다 . 워치독은 Netlet 프록시 포트를 모니터링하여 프록시가 다운되면 표시합니다 .

- Netlet 프록시를 수동으로 시작합니다 .

  netlet-proxy-install-root/SUNWps/bin/netletd -n *gateway-profile-name* start

  여기서 *gateway-profile-name* 은 필요한 게이트웨이 인스턴스에 해당하는 프로필 이름입니다 .

➤ **Netlet 프록시 워치독을 구성하려면**

워치독이 Netlet 프록시의 상태를 모니터하는 시간 간격을 구성할 수 있습니다. 시간 간격은 기본적으로 60 초로 설정됩니다 . 이 설정을 변경하려면 crontab 에서 다음 라인을 편집합니다 .

```
0-59 * * * * netlet-install-dir/bin/checkgw /var/opt/SUNWps/.gw 5> /dev/null 2>&1
```

Rewriter 프록시 사용

Rewriter 프록시는 인트라넷에 설치됩니다 . 게이트웨이는 컨텐트를 직접 검색하는 대신 , 컨텐트를 가져와 게이트웨이로 반환하는 Rewriter 프록시로 모든 요청을 전달합니다 .

Rewriter 프록시 사용을 통해 얻을 수 있는 이점은 2 가지입니다 .

- 게이트웨이와 서버 사이에 방화벽이 있는 경우 방화벽에서는 포트를 2 개만 열면 됩니다 . 하나는 게이트웨이와 Rewriter 프록시 사이의 포트이고 다른 하나는 게이트웨이와 Portal Server 사이의 포트입니다 .
- 대상 서버가 HTTP 프로토콜 (HTTPS 아님) 만 지원하고 있더라도 이제 게이트웨이와 인트라넷 사이의 HTTP 트래픽이 안정적으로 됩니다 .

Rewriter 프록시를 지정하지 않으면 게이트웨이 구성 요소에서 사용자가 인트라넷 컴퓨터에 액세스하려고 할 때 인트라넷 컴퓨터에 직접 연결을 구성합니다 .

Rewriter 프록시를 로드 조정기로 사용하는 경우에는 Rewriter 의 platform.conf.*instance_name* 이 로드 조정기 URL 을 가리키는지 확인해야 합니다 . Portal Servers 목록에 로드 조정기 호스트가 지정되어 있는지도 확인해야 합니다 .

각 게이트웨이 인스턴스에 대해 여러 개의 Rewriter 프록시 인스턴스를 가진 경우에는 (포털 노드에 있을 필요는 없음) Rewrite 프록시의 단일 포트가 아닌 platform.conf 에서 각 Rewriter 프록시의 세부 정보를 *host-name:port* 와 같은 형식으로 입력합니다 .

Rewriter 프록시의 인스턴스 만들기

Portal Server 노드에 Rewriter 프록시의 새 인스턴스를 만들려면 `rwpmultiinstance` 스크립트를 사용합니다 . 게이트웨이 프로필을 만든 후에 이 스크립트를 실행하는 것이 좋습니다 .

1. 루트로 로그인하여 다음 디렉토리로 찾아 갑니다 .

   rewriter-proxy-install-root/SUNWps/bin

2. 다중 인스턴스 스크립트를 실행합니다 .

   ./rwpmultiinstance

3. 스크립트에 나타나는 질문에 답합니다 .

   - 새 rewriter 프록시 인스턴스의 이름은 무엇입니까 ?
   - 이 노드에 같은 이름으로 구성된 rewriter 프록시 인스턴스가 있으면 이 rewriter 프록시 인스턴스에도 같은 구성을 사용할 것인지 묻는 메시지가 나타납니다 .
   - 예라고 답한 경우 , 다음 두 질문에 답하십시오 .
     - 새 rewriter 프록시 인스턴스에서는 어떤 포트를 수신합니까 ?
     - 설치 후 rewriter 프록시를 시작하시겠습니까 ?
   - 아니오라고 답한 경우 , 다음 질문에 답하십시오 .
     - 새 rewriter 프록시 인스턴스에서는 어떤 프로토콜을 사용합니까 ?
     - 새 rewriter 프록시 인스턴스에서는 어떤 포트를 수신합니까 ?
     - 조직의 이름은 무엇입니까 ?
     - 부서 이름은 무엇입니까 ?
     - 구 / 군 / 시의 이름은 무엇입니까 ?
     - 시 / 도의 이름은 무엇입니까 ?
     - 2 자로 된 국가 번호는 무엇입니까 ?
     - 인증서 데이터베이스 비밀번호는 무엇입니까 ?
     - 사용자 로그인 비밀번호는 무엇입니까 ?
     - 관리 콘솔에 새 게이트웨이 프로필을 만들었습니까 ?
     - 예라고 답한 경우 , 설치 후 rewriter 프록시를 시작하시겠습니까 ?

4. 새 게이트웨이 프로필 이름으로 rewriter 프록시의 새 인스턴스를 시작합니다 .

   rewriter-proxy-install-root/SUNWps/bin/rwproxyd -n *gateway-profile-name* start

   여기서 *gateway-profile-name* 은 필요한 게이트웨이 인스턴스에 해당하는 프로필 이름입니다 .

Rewriter 프록시 사용 설정

Identity Server 관리 콘솔의 SRA 구성에서 게이트웨이 서비스를 통해 Rewriter 프록시를 활성화합니다 . 241 페이지의 "Rewriter 프록시 목록 사용과 만들기 " 를 참조하십시오 .

Rewriter 프록시 다시 시작

프록시가 예기치 않게 중단될 때마다 다시 시작하도록 Rewriter 프록시를 구성할 수 있습니다 . 모니터할 워치독 프로세스를 예약하고 이 경우가 발생했을 때 다시 시작할 수 있습니다 .

Rewriter 프록시를 수동으로 다시 시작할 수도 있습니다 .

➤ Rewriter 프록시를 다시 시작하려면

단말기 창에서 루트로 연결하고 다음 작업 중 하나를 수행합니다 .

- 워치독 프로세스를 시작합니다 .

  rewriter-proxy-install-root`/SUNWps/bin/rwproxd watchdog on`

  그러면 crontab 에 항목이 만들어지고 워치독 프로세스가 활성 상태가 됩니다 . 워치독은 포트를 모니터링하여 프록시가 다운되면 표시합니다 .

- 수동으로 시작합니다 .

  rewriter-proxy-install-root`/SUNWps/bin/rwproxd -n` *gateway-profile-name* `start`

  여기서 *gateway-profile-name* 은 필요한 게이트웨이 인스턴스에 해당하는 프로필 이름입니다 .

➤ Rewriter 프록시 워치독을 구성하려면

워치독이 Rewriter 프록시 상태를 모니터링하는 시간 간격을 구성할 수 있습니다 . 시간 간격은 기본적으로 60 초로 설정됩니다 . 이 설정을 변경하려면 crontab 에서 다음 라인을 편집합니다 .

```
0-59 * * * * rewriter-proxy-install-root/bin/checkgw /var/opt/SUNWps/.gw 5>
/dev/null 2>&1
```

게이트웨이에서 역 프록시 사용

프록시 서버는 인트라넷에 인터넷 컨텐트를 서비스하고 역 프록시는 인터넷에 인트라넷 컨텐트를 서비스합니다. 인터넷 컨텐트를 제공하며 로드 균형 조정과 캐싱을 수행하도록 역 프록시를 배포할 수 있습니다.

이 배포에서 게이트웨이 전방에 타사의 역 프록시가 사용된다면 게이트웨이의 URL 대신 역 프록시의 URL로 응답을 다시 써야 합니다. 이를 위해 다음 구성이 필요합니다.

➤ **역 프록시를 활성화하려면**

1. 루트로 로그인하여 필요한 게이트웨이 인스턴스의 `platform.conf` 파일을 편집합니다.

   `/etc/opt/SUNWps/platform.conf.`*gateway-profile-name*

2. 다음 항목을 추가합니다.

   `gateway.virtualhost=`*fully-qualified-gateway-host gateway-ip-address fully-qualified-reverse-proxyhost*

   `gateway.enable.customurl=true` (이 값은 기본적으로 false로 설정됩니다.)

   `gateway.httpurl=`*http reverse-proxy-URL*

   `gateway.httpsurl=`*https reverse-proxy-URL*

   `gateway.httpurl`은 게이트웨이 프로필에서 HTTP 포트로 나열된 포트에서 수신된 요청에 대한 응답을 다시 쓰는데 사용됩니다.

   `gateway.httpsurl`은 게이트웨이 프로필에서 HTTPS 포트로 나열된 포트에서 수신된 요청에 대한 응답을 다시 쓰는데 사용됩니다.

3. 게이트웨이를 다시 시작합니다.

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

값이 지정되어 있지 않으면 게이트웨이에서는 일반 작동으로 기본값을 설정합니다.

클라이언트 정보 가져오기

게이트웨이에서 클라이언트 요청을 임의 서버로 전달할 때 HTTP 헤더를 HTTP 요청에 추가합니다. 이 헤더를 사용하여 추가 클라이언트 정보를 가져오고 게이트웨이가 있는지 감지할 수 있습니다.

HTTP 요청 헤더를 보려면 `platform.conf` 파일의 항목을 `gateway.error=message` 로 설정한 다음 servlet API 에서 request.getHeader() 를 사용합니다 . 다음 표에는 HTTP 헤더에 있는 정보가 나열되어 있습니다 .

표 2-3 HTTP 헤더의 정보

헤더 .	구문	설명
PS-GW-PDC	X-PS-GW- PDC: true/false	게이트웨이에서 PDC 의 사용 설정 여부를 나타냅니다 .
PS-Netlet	X-PS-Netlet:enabled=true/false	게이트웨이에서 Netlet 의 사용 설정 여부를 나타냅니다 . 사용 설정된 경우 암호화 옵션이 채워져서 게이트웨이가 HTTPS(encryption=ssl) 또는 HTTP 모드 (encryption=plain) 중 어느 쪽에서 실행 중인지 보여줍니다 . 예 : PS-Netlet: enabled=false Netlet 이 사용 해제되었습니다 . PS-Netlet: enabled=true; encryption=ssl 게이트웨이가 SSL 모드에서 실행되며 Netlet 이 사용 설정되었습니다 . Netlet 이 활성화되지 않은 경우에는 encryption=ssl/plain 이 채워지지 않습니다 .
PS-GW-URL	X-PS-GW-URL: http(s)://gatewayURL(:port)	클라이언트가 연결된 URL 을 나타냅니다 . 비표준 포트인 경우 (즉 , 포트 80/443 이 아닌 상태로 게이트웨이가 HTTP/HTTPS 모드에 잇는 경우) ":port" 도 채워집니다 .

표 2-3 HTTP 헤더의 정보

헤더 .	구문	설명
PS-GW-Rewriting-URL	X-PS-GW-URL: http(s)://gatewayURL(:port)/[SessionInfo]	게이트웨이가 모든 페이지를 다시 쓰는 URL 을 나타냅니다 . 1. 브라우저에서 쿠키를 지원하는 경우 이 헤더 값은 PS-GW-URL 헤더와 같습니다 . 2. 브라우저가 쿠키를 지원하지 않고 • 대상 호스트가 " 사용자 세션 쿠키가 전달될 사용자 세션 " 필드에 있으면 값은 게이트웨이가 페이지를 쓰는 실제 URL 이 됩니다 (암호화된 세션 ID 정보 포함). • 또는 대상 호스트가 " 사용자 세션 쿠키가 전달될 사용자 세션 " 필드에 없으면 세션 ID 문자열은 "$SessionID" 가 됩니다 . 참고 : 응답의 일부로 사용자의 Identity Server sessionId 가 변경되면 (인증 페이지에서 오는 응답과 같이) 페이지는 이전에 헤더에 표시된 값이 아닌 그 값으로 다시 쓰여집니다 . 예 : • 브라우저에서 쿠키를 지원하는 경우 PS-GW-Rewriting-URL: https://siroe.india.sun.com:10443/ • 브라우저에서 쿠키를 지원하지 않지만 endserver 가 " 사용자 세션 쿠키가 전달될 사용자 세션 " 필드에 있는 경우 PS-GW-Rewriting-URL: https://siroe.india.sun.com:10443/SessIDValCustomEncodedValue/ • 브라우저에서 쿠키를 지원하지 않고 endserver 가 " 사용자 세션 쿠키가 전달될 사용자 세션 " 필드에 없는 경우 PS-GW-Rewriting-URL: https://siroe.india.sun.com:10443/$SessionID
PS-GW-CLientIP	X-PS-GW-CLientIP: *IP*	게이트웨이가 recievedSocket.getInetAddress().getHostAddress() 로부터 가져온 IP 입니다 . 이 IP 는 게이트웨이에 직접 연결되면 클라이언트의 IP 가 됩니다 . 참고 : JSS/NSS 버그로 인해 현재 존재하지는 않습니다 .

인증 체이닝 사용

인증 체이닝은 인증의 일반 메커니즘에서 보안을 한층 높은 수준으로 강화합니다 . 사용자가 2 개 이상 인증 메커니즘에 대해 인증 받도록 설정할 수 있습니다 .

여기에 설명된 절차는 게이트웨이에서 개인 디지털 인증서 (PDC) 인증과 함께 인증 체이닝을 사용하는 경우에만 적용됩니다 . 게이트웨이에서 PDC 인증을 사용하지 않는 인증 체이닝에 대한 내용은 *Identity Server 관리 설명서*를 참조하십시오 .

예를 들어 , PDC, Unix 및 Radius 인증 모듈을 체인 연결한 경우에는 사용자가 표준 포털 데스크탑에 액세스하려면 이 3 개 모듈에 대한 인증을 모두 거쳐야 합니다 .

참고 PDC 는 사용 설정된 경우 사용자에게 항상 가정 먼저 제시되는 인증 모듈입니다 .

➤ 기존 PDC 인스턴스에 인증 모듈을 추가하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. 필요한 조직을 선택합니다 .
3. [보기] 드롭다운 메뉴에서 [서비스] 를 선택합니다 .

   왼쪽 창에 서비스가 표시됩니다 .
4. [인증 구성] 옆의 화살표를 클릭합니다 .

   서비스 인스턴스 목록이 표시됩니다 .
5. gatewaypdc 를 클릭합니다 .

   Gatewaypdc 속성 페이지가 표시됩니다 .
6. [인증 구성] 앞의 [편집] 을 클릭합니다 .

   [모듈 추가] 가 나타납니다 .
7. [모듈 이름] 을 선택하고 [플래그] 를 [필요] 로 설정합니다 . 빈 칸으로 남겨둬도 됩니다 .
8. [확인] 을 클릭합니다 .
9. 모듈을 하나 이상 추가한 다음 [저장] 을 클릭합니다 .
10. gatewaypdc 속성 페이지에서 [저장] 을 클릭합니다 .

11. 변경 내용을 적용하려면 게이트웨이를 다시 시작합니다 .

    gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

와일드카드 인증 사용

와일드카드 인증에서는 완전한 정규 DNS 호스트 이름에 와일드카드 문자가 있는 단일 인증을 수락합니다 .

그러면 같은 도메인에서 여러 호스트에게 인증을 허용할 수 있습니다 . 예를 들어 , *.domain.com 에 대한 인증을 abc.domain.com 및 abc1.domain.com 에 사용할 수 있습니다 . 사실 이 인증은 domain.com 도메인에 있는 모든 호스트에 유효합니다 .

완전한 정규 호스트 이름에 * 를 지정해야 합니다 . 예를 들어 , 완전한 정규 호스트 이름이 abc.florizon.com 인 경우 이 이름을 *.florizon.com 으로 지정하십시오 . 이제 생성된 인증서가 florizon.com 도메인에 있는 모든 호스트 이름에 유효합니다 .

브라우저 캐싱 사용 해제

게이트웨이 구성 요소는 웹 브라우저를 사용하여 어느 위치에서든 백엔드 기업 데이터에 안전하게 액세스하므로 클라이언트에 의해 정보가 로컬로 캐싱되지 않아야 합니다 .

특정 게이트웨이의 platform.conf 파일에 있는 속성을 수정하여 게이트웨이를 통해 리디렉션된 페이지의 캐싱을 사용 해제할 수 있습니다 .

이 옵션을 사용 해제하면 게이트웨이 성능에 영향이 있을 수 있습니다 . 표준 포털 데스크탑을 새로 고칠 때마다 게이트웨이는 브라우저에서 이전에 캐싱한 이미지와 같이 페이지에서 참조되는 모든 항목을 검색해야 합니다 . 그러나 이 기능을 사용 설정하면 원격 액세스 보안 컨텐트가 클라이언트 사이트에 캐싱된 풋프린트를 남기지 않습니다 . 기업 네트워크가 인터넷 카페에서 또는 기업 IT 제어를 받지 않는 유사한 원격 위치에서 액세스되는 경우 이 이점은 성능상의 불이익 보다 훨씬 큽니다 .

➤ 브라우저 캐싱을 사용 해제하려면

1. 루트로 로그인하여 필요한 게이트웨이 인스턴스의 platform.conf 파일을 편집합니다 .

   /etc/opt/SUNWps/platform.conf.*gateway-profile-name*

2. 다음 라인을 편집합니다 .

   gateway.allow.client.caching=true

   이 값은 기본적으로 true 로 설정되어 있습니다 . 값을 false 로 변경하여 클라이언트 쪽에서 브라우저 캐싱을 사용 해제합니다 .

3. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

게이트웨이 서비스 사용자 인터페이스 사용자 정의

이 부분에서는 편집할 수 있는 여러 속성 파일에 대해 설명합니다 .

srapGateway.properties 파일

다음과 같은 목적으로 이 파일을 편집할 수 있습니다 .

- 게이트웨이가 실행 중에 나타날 수 있는 오류 메시지를 사용자 정의할 때 .
  - HTML-CharSets=ISO-8859-1 은 이 파일을 만드는 데 사용되는 문자 집합을 지정합니다 .
  - 중괄호 안의 숫자(예 : {0})는 값이 런타임으로 표시된다는 것을 뜻합니다 . 필요에 따라 이 숫자와 연관된 레이블을 변경하거나 레이블을 재배열할 수 있습니다 . 레이블 숫자와 오류는 연관되어 있기 때문에 레이블이 표시되는 메시지와 상응하는지 확인하십시오 .
- 로그 정보를 사용자 정의할 때 .

  기본적으로 srapGateway.properties 파일은 *portal-server-install-root*/SUNWps/locale 디렉토리에 있습니다 . 게이트웨이 컴퓨터에 나타나는 모든 메시지 (게이트웨이 관련 메시지) 는 메시지의 언어와는 상관 없이 이 파일에 있습니다 .

  클라이언트 표준 포털 데스크탑에 나타나는 메시지의 언어를 변경해야 하는 경우 이 파일을 각 로켈 디렉토리로 복사합니다 (예 : *portal-server-install-root*/SUNWps/locale_en_US).

srapgwadminmsg.properties 파일

다음과 같은 이유로 이 파일을 편집할 수 있습니다 .

- 관리 콘솔의 게이트웨이 서비스에 대한 버튼에 나타나는 레이블을 사용자 정의할 때 .
- 게이트웨이를 구성하는 중에 나타나는 상태 메시지 및 오류 메시지를 사용자 정의할 때 .

연합 관리 사용

연합 관리를 사용하면 사용자가 하나의 네트워크 아이디를 가질 수 있도록 로컬 아이디를 집계할 수 있습니다 . 연합 관리에서는 네트워크 아이디를 사용하여 사용자가 한 서비스 공급자의 사이트에 로그인할 경우 아이디를 재인증 받지 않고도 다른 서비스 공급자의 사이트에 액세스할 수 있도록 해 줍니다. 이를 단일 사인온이라 합니다.

연합 관리는 Portal Server 에서 개방 모드 및 보안 모드로 구성할 수 있습니다 . *Portal Server 관리 설명서*에서는 개방 모드로 연합 관리를 구성하는 방법에 대해 설명합니다 . 연합 관리를 보안 원격 액세스를 사용하여 보안 모드에서 구성하려면 개방 모드에서 올바로 작동하는지 확인해야 합니다 . 사용자가 같은 브라우저에서 개방 모드와 보안 모드 모두에서 연합 관리를 사용할 수 있도록 하려면 쿠키를 지우고 브라우저로부터 캐싱해야 합니다 .

연합 관리에 대한 자세한 내용은 *Identity Server Customization and API Guide* 를 참조하십시오 .

연합 관리 시나리오

사용자가 최초 서비스 공급자에게 인증을 받습니다 . 서비스 공급자는 웹 기반 서비스를 제공하는 상업적 조직이거나 비영리 조직을 말합니다 . 이렇게 넓은 범주에는 인터넷 포털 , 대리점 , 운송 공급자 , 금융 기관 , 엔터테인먼트 회사 , 도서관 , 대학 및 정부 기관이 모두 포함될 수 있습니다 .

서비스 공급자는 쿠키를 사용하여 클라이언트 브라우저에 사용자의 세션 정보를 저장합니다 . 쿠키에도 사용자의 아이디 공급자가 포함될 수 있습니다 .

아이디 공급자는 인증 서비스를 전문적으로 제공하는 서비스 공급자를 말합니다 . 인증을 위한 관리 서비스로 아이디 공급자는 아이디 정보를 유지 관리하기도 합니다 . 아이디 공급자에 의해 허가된 인증은 제휴 관계에 있는 모든 서비스 공급자에게 유효합니다 .

사용자가 아이디 공급자와 제휴되지 않은 서비스에 액세스하려고 하면 아이디 공급자는 쿠키를 비제휴 서비스 공급자에게 전달합니다 . 그리고 나면 이 서비스 공급자가 쿠키에 명명된 아이디 공급자에게 액세스할 수 있습니다 .

그러나 쿠키는 여러 DAN 도메인에서 읽을 수 없기 때문에 서비스 공급자를 올바른 아이디 공급자에게 리디렉션하여 사용자에게 단일 사인온이 가능하도록 공용 도메인 쿠키 서비스를 사용합니다 .

연합 관리 리소스 구성

연합 자원 , 서비스 공급자 , 아이디 공급자 , 아이디 공급자 및 공용 도메인 쿠키 서비스 (CDCS) 는 상주해 있는 위치를 기준으로 게이트웨이 프로필에 구성합니다 . 이 부분에서는 3 가지 시나리오를 구성하는 방법에 대해 설명합니다 .

1. 모든 리소스가 기업 인트라넷 안에 있는 경우 .
2. 모든 리소스가 기업 인트라넷에 있지 않거나 아이디 공급자가 인터넷에 상주하는 경우 .
3. 모든 리소스가 기업 인트라넷에 있지 않거나 서비스 공급자는 인터넷에 상주하는 타사이고 아이디 공급자는 게이트웨이에서 보호되는 경우 .

구성 1

이 구성에서는 서비스 공급자 , 아이디 공급자 및 공용 도메인 쿠키 서비스가 같은 기업 인트라넷에 배치되고 아이디 공급자는 인터넷 DNS (Domain Name Server) 에 게시되지 않습니다 . CDCS 는 선택 사항입니다 .

이 구성에서는 게이트웨이가 Portal Server 가 되는 서비스 공급자를 지정합니다 . 이 구성은 Portal Server 의 다중 인스턴스에 유효합니다 .

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. 관리 콘솔에서 [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [쿠키 관리 사용] 확인란을 선택하고 쿠키 관리의 사용을 설정합니다 .

7. Portal Server 필드로 스크롤하고 Portal Server 이름을 입력하여 인증되지 않은 URL 목록에 나열된 /amserver 또는 /portal/dt 등의 관련 URL 을 사용할 수 있도록 합니다 . 예 :

   `http://`*idp-host:port*`/amserver/js`

   `http://`*idp-host:por*`t/amserver/UI/Login`

   `http://`*idp-host:port*`/amserver/css`

   `http://`*idp-host:port*`/amserver/SingleSignOnService`

   `http://`*idp-host:port*`/amserver/UI/blank`

   `http://`*idp-host:port*`/amserver/postLogin`

   `http://`*idp-host:port*`/amserver/login_images`

8. Portal Server 필드로 스크롤하여 Portal Server 이름을 입력합니다 . 예를 들어 /amserver 를 입력합니다 .
9. [저장] 을 누릅니다 .
10. [보안] 탭을 클릭합니다 .
11. 인증되지 않은 URL 목록으로 스크롤하여 연합 리소스를 추가합니다 . 예 :

    `/amserver/config/federation`

    `/amserver/IntersiteTransferService`

    `/amserver/AssertionConsumerservice`

    `/amserver/fed_images`

    `/amserver/preLogin`

    `/portal/dt`

12. [추가] 를 누릅니다 .
13. [저장] 을 누릅니다 .
14. 웹 프록시에서 인증되지 않은 URL 목록에 나열된 URL 에 접속이 필요하면 [프록시] 탭을 클릭합니다 .
15. [도메인 및 부속 도메인의 프록시] 필드로 스크롤하여 필요한 웹 프록시를 입력합니다 .
16. [추가] 를 누릅니다 .
17. [저장] 을 누릅니다 .

18. 단말기 창에서 게이트웨이를 다시 시작합니다 .

    gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

구성 2

이 구성에서는 아이디 공급자 , 아이디 공급자 및 공용 도메인 쿠키 공급자 (CDCP) 가 같은 기업 인트라넷에 배치되지 *않았거나* 아이디 공급자가 인터넷에 상주하는 타사 공급자입니다 .

이 구성에서는 게이트웨이가 Portal Server 가 되는 서비스 공급자를 지정합니다 . 이 구성은 Portal Server 의 다중 인스턴스에 유효합니다 .

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. 관리 콘솔에서 [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [쿠키 관리 사용] 확인란을 선택하고 쿠키 관리의 사용을 설정합니다 .
7. Portal Server 필드로 스크롤하고 서비스 공급자 Portal Server 이름을 입력하여 인증되지 않은 URL 목록에 나열된 /amserver 또는 /portal/dt 등의 관련 URL 을 사용할 수 있도록 합니다 .

   http://*idp-host:port*/amserver/js

   http://*idp-host:port*/amserver/UI/Login

   http://*idp-host:port*/amserver/css

   http://*idp-host:port*/amserver/SingleSignOnService

   http://*idp-host:port*/amserver/UI/blank

   http://*idp-host:port*/amserver/postLogin

   http://*idp-host:port*/amserver/login_images
8. [저장] 을 누릅니다 .
9. [보안] 탭을 클릭합니다 .

10. 인증되지 않은 URL 목록으로 스크롤하여 연합 리소스를 추가합니다 . 예 :

    ```
    /amserver/config/federation

    /amserver/IntersiteTransferService

    /amserver/AssertionConsumerservice

    /amserver/fed_images

    /amserver/preLogin

    /portal/dt
    ```

11. [추가] 를 누릅니다 .
12. [저장] 을 누릅니다 .
13. 웹 프록시에서 인증되지 않은 URL 목록에 나열된 URL 에 접속이 필요하면 [프록시] 탭을 클릭합니다 .
14. [도메인 및 부속 도메인의 프록시] 필드로 스크롤하여 필요한 웹 프록시를 입력합니다 .
15. [추가] 를 누릅니다 .
16. [저장] 을 누릅니다 .
17. 단말기 창에서 게이트웨이를 다시 시작합니다 .

    gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

구성 3

이 구성에서는 아이디 공급자 , 아이디 공급자 및 공용 도메인 쿠키 공급자 (CDCP) 가 같은 기업 인트라넷에 배치되지 *않았거나* 서비스 공급자가 인터넷에 상주하는 타사이고 아이디 공급자는 게이트웨이에 의해 보호됩니다 .

이 구성에서는 게이트웨이가 Portal Server 가 되는 아이디 공급자를 지정합니다 .

이 구성은 Portal Server 의 다중 인스턴스에 유효합니다 . 이 구성은 인터넷에서는 구현되는 경우가 거의 없지만 어떤 기업 네트워크에는 인트라넷에 이러한 구성이 있을 수 있습니다 . 즉 , 아이디 공급자는 방화벽으로 보호되는 서브 네트에 있고 서비스 공급자는 기업 네트워크 내에서 직접 액세스 가능한 경우를 말합니다 .

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. 관리 콘솔에서 [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [쿠키 관리 사용] 확인란을 선택하고 쿠키 관리의 사용을 설정합니다 .
7. Portal Server 필드로 스크롤하고 아이디 공급자 Portal Server 이름을 입력하여 인증되지 않은 URL 목록에 나열된 /amserver 또는 /portal/dt 등의 관련 URL 을 사용할 수 있도록 합니다 .

   `http://`*idp-host:port*`/amserver/js`

   `http://`*idp-host:port*`/amserver/UI/Login`

   `http://`*idp-host:port*`/amserver/css`

   `http://`*idp-host:port*`/amserver/SingleSignOnService`

   `http://`*idp-host:port*`/amserver/UI/blank`

   `http://`*idp-host:port*`/amserver/postLogin`

   `http://`*idp-host:port*`/amserver/login_images`
8. [저장] 을 누릅니다 .
9. [보안] 탭을 클릭합니다 .
10. 인증되지 않은 URL 목록으로 스크롤하여 연합 리소스를 추가합니다 . 예 :

    `/amserver/config/federation`

    `/amserver/IntersiteTransferService`

    `/amserver/AssertionConsumerservice`

    `/amserver/fed_images`

    `/amserver/preLogin`

    `/portal/dt`
11. [추가] 를 누릅니다 .
12. [저장] 을 누릅니다 .

13. 웹 프록시에서 인증되지 않은 URL 목록에 나열된 URL 에 접속이 필요하면 [프록시] 탭을 클릭합니다 .
14. [도메인 및 부속 도메인의 프록시] 필드로 스크롤하여 필요한 웹 프록시를 입력합니다 .
15. [추가] 를 누릅니다 .
16. [저장] 을 누릅니다 .
17. 단말기 창에서 게이트웨이를 다시 시작합니다 .

    gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

3 장

Proxylet 및 Rewriter

이 장에서는 Proxylet 과 Rewriter 에 대해 설명합니다 . 이 구성 요소는 사용자가 게이트웨이를 통해 인트라넷 웹 페이지에 액세스할 수 있게 해 줍니다 . 이 작업은 여러 가지 방법으로 수행합니다 . Proxylet 은 Rewriter 처럼 웹 페이지를 구문 분석하지는 않습니다 .

Proxylet 에 대해서는 다음과 같은 주제를 다룹니다 .

- Proxylet 개요

Rewriter 에 대해서는 다음과 같은 주제를 다룹니다 .

- Rewriter 개요
- 문자 집합 암호화
- Rewriter 사용 시나리오
- 규칙 집합 작성
- 공용 인터페이스 (규칙 집합 DTD)
- 게이트웨이 서비스에서 Rewriter 구성
- 디버깅 로그 사용의 문제 해결
- 공용 인터페이스 (규칙 집합 DTD)
- 작업 예제
- 사례 연구
- 6.x 규칙 집합을 3.0 과 매핑

Proxylet 개요

Proxylet 은 클라이언트 시스템에서 실행되는 동적 프록시 서버입니다 . Proxylet 은 URL 을 게이트웨이로 리디렉션합니다 . 이 때 클라이언트 시스템에 있는 브라우저의 프록시 설정을 읽고 로컬 프록시 서버 또는 Proxylet 을 가리키도록 수정합니다 .

여기서는 HTTP 와 SSL 을 모두 지원하며 게이트웨이의 전송 모드를 상속합니다 . 게이트웨이가 SSL 에서 실행되도록 구성되어 있으면 SSL, Proxylet 은 클라이언트 시스템과 게이트웨이 사이에 안전한 채널을 구성합니다 . 클라이언트 JVM 이 1.4 이상이거나 필요한 jar 파일이 클라이언트 시스템에 상주하는 경우에는 Proxylet 에서 JSSE API 를 사용합니다 . 그렇지 않으면 KSSL API 를 사용합니다 .

게이트웨이로 리디렉션되는 URL 의 도메인 및 부속 도메인은 게이트웨이 프로필에서 지정됩니다 . URL 이 게이트웨이가 처리하는 도메인에 속하지 않으면 요청은 인터넷으로 지정됩니다 . 게이트웨이 프로필에 특정 URL 도메인이 나열되어 있으면 Proxylet 은 게이트웨이를 가리키도록 클라이언트 프록시 설정을 재설정합니다 .

Proxylet 은 클라이언트 IP 주소와 포트를 지정한 Identity Server 관리 콘솔에서 활성화됩니다 . Proxylet 을 활성화하면 클라이언트 시스템에서 다음 사항이 확인됩니다 .

- 정확한 브라우저 권한
- JVM 버전 2(Netscape 브라우저의 경우)
- 브라우저가 Netscape 7.0, Mozilla 1.4.1, Internet Explorer 5.0 이상인지 여부
- 컴퓨터 또는 장치가 서버 응용 프로그램을 실행할 수 있는지 여부

요구 사항이 충족되면 클라이언트 시스템에서 작은 애플릿을 다운로드하여 시작합니다 . 사용자의 컴퓨터에 JRE 1.3.1 이상이 설치되어 있지 않으면 Proxylet 과 함께 JRE 가 자동으로 다운로드됩니다 .

여기서는 프록시 자동 구성 (PAC) 파일을 사용하는 경우 이 파일로부터 프록시 설정을 검색하거나 , 또는 프록시 구성 목록에서 검색합니다 .

Proxylet 사용의 이점

Rewriter 와 달리 Proxylet 은 설치 후 변경이 거의 또는 전혀 없는 완벽한 솔루션입니다 . Microsoft Exchange Server 등의 타사 소프트웨어와 쉽게 통합할 수 있습니다 . Proxylet 은 웹 컨텐트를 다루지 않기 때문에 게이트웨이의 성능도 좋아집니다 .

Proxylet 구성

Proxylet 의 활성화와 구성에 관한 자세한 내용은 321 페이지 12 장 "Proxylet 구성 " 을 참조하십시오 .

Rewriter 개요

SRA 의 Rewriter 구성 요소를 통해 최종 사용자가 게이트웨이를 가리키도록 웹 페이지의 URI(Uniform Resource Identifier) 를 수정하여 인트라넷을 찾아볼 수 있습니다. URI 는 등록된 이름 공간에서 이름을 캡슐화하여 여기에 이름 공간으로 이름을 붙이는 방법을 정의합니다 . 가장 일반적인 URI 형태는 URL (Uniform Resource Locator) 입니다 . URL 은 http, ftp, mailto, file 및 news 와 같은 다양한 프로토콜을 가질 수 있습니다 .

RFC-1738 에 규정되어 있으며 HTTP 또는 HTTPS 의 프로토콜을 갖는 모든 표준 URL 은 Rewriter 에 의해 인식되어 다시 작성됩니다 . 프로토콜은 대소문자를 구별하지 않습니다 . 예를 들어 hTtP, HTtp 및 httP 모두 올바른 표현입니다 . 다음은 URL 의 몇 가지 예입니다 .

```
http://www.my.work.com/

http://www.w3.org:8000/imaginary/test

http://www.myu.edu/org/admin/people#andy

http://info.my.org/AboutUs/Index/Phonebook?dobbins

http://www.w3.org/RDB/EMP?where%20name%3Ddobbins

http://info.my.org/AboutUs/Phonebook

http://user:password@abc.com
```

Rewriter 는 Internet Explore 와 Netscape 에서 지원하는 일부 기본적인 표준 이외 URL 의 재작성도 지원합니다 . 표준 이외 URL 을 표준 형식으로 변환할 때 필요한 정보는 URL 이 표시되는 페이지의 기본 URL 에서 가져옵니다 . 이 정보에는 다음이 포함됩니다 .

- protocol
- 호스트 이름
- 포트
- path

Rewriter 는 상대 URL 의 일부인 경우에만 백슬래시를 지원합니다 .

예 :

`http://abc.sesta.com\index.html` 은 다시 작성됩니다 .

다음 URL 은 다시 작성되지 않습니다 .

`http:\\abc.sesta.com`.

`http:/abc.com`

문자 집합 암호화

HTTP 표준에 따르면 HTTP 헤더 또는 HTML 메타 태그에서 웹 페이지에 사용할 문자 집합을 지정해야 합니다 . 하지만 이 정보를 사용할 수 없는 경우도 있습니다 . 데이터의 암호화를 설정하고 만든 사람의 의도에 맞게 데이터를 표시하려면 문자 집합을 알아야 합니다 .

Sun Microsystems 에서는 문자 집합 검색에 사용되는 타사 제품을 제공합니다 . 이 제품을 사용하려면 SUNWjchdt 패키지를 설치해야 합니다 . 제품이 설치되면 Rewriter 에서 필요한 경우 검색하여 사용합니다 .

참고 이 제품을 사용하면 성능이 저하될 수 있기 때문에 필요한 경우에만 설치해야 합니다 . 설치 , 구성 및 사용에 관한 자세한 내용은 `jcharset_readme.txt` 를 참조하십시오 .

Rewriter 사용 시나리오

사용자가 게이트웨이를 통해 인트라넷 웹 페이지에 액세스하려고 할 때 Rewriter 가 웹 페이지를 사용할 수 있도록 해줍니다 . 다음 구성 요소에서 Rewriter 를 사용합니다 .

- URLScraper
- 게이트웨이

URLScraper

URL Scraper 제공자는 구성된 URI 에서 컨텐트를 가져와 브라우저로 보내기 전에 모든 상대 URI 를 절대 URI 로 확장합니다 .

예를 들어 , 사용자가 다음과 같은 컨텐트를 가진 사이트에 액세스하려고 하는 경우

```
<a href="../mypage.html">
```

Rewriter 는 이것을 다음으로 변환합니다 .

```
<a href="http://yahoo.com/mypage.html">
```

여기서 `http://yahoo.com/test/` 는 페이지의 기본 URL 입니다 .

URLScraper 공급자에 대한 자세한 내용은 *Portal Server 관리 설명서*를 참조하십시오 .

게이트웨이

게이트웨이는 인터넷 포털에서 컨텐트를 가져와 이를 브라우저로 보내기 전에 브라우저로부터의 이후 URI 요청이 게이트웨이로 올 수 있도록 기존 URI 에 게이트웨이 URI 접두어를 덧붙입니다 .

예를 들어 , 다음과 같은 컨텐트를 갖는 인터넷 컴퓨터의 HTML 페이지에 액세스하려고 하는 사용자에 대해

```
<a href="http://mymachine.intranet.com/mypage.html>"
```

Rewriter 는 이 URL 에 다음과 같이 게이트웨이에 대한 참조를 갖는 URL 이름을 접두어로 붙입니다 .

```
<a href="https://gateway.company.com/http://mymachine.intranet.com/
mypage.html>"
```

사용자가 이 앵커와 관련된 링크를 누르면 브라우저가 게이트웨이에 접속합니다 . 게이트웨이는 `mymachine.intranet.com` 으로부터 `mypage.html` 의 컨텐트를 가져옵니다 .

게이트웨이는 가져온 웹 페이지에서 다시 작성할 요소를 결정하기 위해 몇 가지 규칙을 사용합니다 .

규칙 집합 작성

[서비스 구성] 탭의 [포털 서버 구성] 부분에서 규칙 집합을 정의합니다 .

규칙 집합에 대한 자세한 내용은 ***Portal Server*** *관리 설명서*를 참조하십시오 . 새 규칙 집합을 만든 후 필요한 규칙을 정의해야 합니다 .

이 부분에서는 다음 주제를 다룹니다 .

- 공용 인터페이스 (규칙 집합 DTD)
- 예제 XML DTD
- 규칙 작성을 위한 절차
- 규칙 집합 가이드라인
- 규칙 집합의 루트 요소 정의
- 재귀적 기능 사용
- HTML 컨텐트에 대한 규칙
- JavaScript 컨텐트에 대한 규칙
- XML 컨텐트에 대한 규칙
- CCS(Cascading Style Sheet) 에 대한 규칙
- WML 에 대한 규칙

공용 인터페이스 (규칙 집합 DTD)

다음은 RuleSet DTD 입니다 .

```
<?xml version="1.0" encoding="UTF-8"?>

<!--

다음 제한은 DTD 에서 표시되지 않지만 프로그래밍에서 고려됩니다 .

    1. 규칙에서 모든 필수 속성은 "*" 일 수 없습니다 .

    2. 아래 요소의 한 인스턴스만 허용되며 순서는 상관 없습니다 .

    1)HTMLRules

    2)JSRules

    3)XMLRules
```

```
    3. 아이디는 항상 소문자여야 합니다 .
-->
<!ENTITY % eURL 'URL'>
<!ENTITY % eEXPRESSION 'EXPRESSION'>
<!ENTITY % eDHTML 'DHTML'>
<!ENTITY % eDJS 'DJS'>
<!ENTITY % eSYSTEM 'SYSTEM'>

<!ENTITY % ruleSetElements '(HTMLRules | JSRules | XMLRules)?'>
<!ENTITY % htmlElements '(Form | Applet | Attribute)*'>
<!ENTITY % jsElements '(Variable | Function)*'>
<!ENTITY % xmlElements '(Attribute | TagText)*'>

<!ELEMENT RuleSet (%ruleSetElements;,%ruleSetElements;,%ruleSetElements;)>
<!ATTLIST RuleSet
    id ID #REQUIRED
    extends CDATA "none"
>

<!-- Rules for identifying rules in HTML content -->
<!ELEMENT HTMLRules (%htmlElements;)>
<!ELEMENT Form EMPTY>
<!ATTLIST Form
    name CDATA #REQUIRED
    field CDATA #REQUIRED
    valuePatterns CDATA ""
    source CDATA "*"
>
```

```
<!ELEMENT Applet EMPTY>
<!ATTLIST Applet
    code CDATA #REQUIRED
    param CDATA "*"
    valuePatterns CDATA ""
    source CDATA "*"
>

<!-- Rules for identifying rules in JS content -->
<!ELEMENT JSRules (%jsElements;)>
<!ELEMENT Variable EMPTY>
<!ATTLIST Variable
    name CDATA #REQUIRED
    type (%eURL; | %eEXPRESSION; | %eDHTML; | %eDJS; | %eSYSTEM;)
"EXPRESSION"
    source CDATA "*"
>

<!ELEMENT Function EMPTY>
<!ATTLIST Function
    name CDATA #REQUIRED
    paramPatterns CDATA #REQUIRED
    type (%eURL; | %eEXPRESSION; | %eDHTML; | %eDJS;) "EXPRESSION"
    source CDATA "*"
>

<!-- Rules for identifying rules in XML content -->
<!ELEMENT XMLRules (%xmlElements;)>
<!ELEMENT TagText EMPTY>
<!ATTLIST TagText
```

```
    tag CDATA #REQUIRED
    attributePatterns CDATA ""
    source CDATA "*"
>

<!ELEMENT Attribute EMPTY>
<!ATTLIST Attribute
    name CDATA #REQUIRED
    tag CDATA "*"
    valuePatterns CDATA ""
    type (%eURL; | %eDHTML; | %eDJS; ) "URL"
    source CDATA "*"
>
```

참고 * 를 규칙 값의 일부로 사용할 수 있습니다 . 그러나 어떤 필수 속성 값도 * 자체가 될 수는 없습니다 . 이러한 규칙은 무시되지만 RuleSetInfo 로그 파일에 메시지가 기록됩니다 . 이 로그 파일에 대한 내용은 136 페이지의 " 디버깅 파일 이름 " 을 참조하십시오 .

예제 XML DTD

이 부분에는 예제 규칙 집합이 들어 있습니다 . 140 페이지의 " 사례 연구 " 를 통해 Rewriter 에서 이러한 규칙을 해석하는 방식을 살펴 볼 수 있습니다 .

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!--
Rules for integrating a mail client with the gateway.
-->
<!DOCTYPE RuleSet SYSTEM "jar://rewriter.jar/resources/RuleSet.dtd">
<RuleSet type="GROUPED" id="owa">
<HTMLRules>
    <Attribute name="action" />
```

```
    <Attribute name="background" />
    <Attribute name="codebase" />
    <Attribute name="href" />
    <Attribute name="src" />
    <Attribute name="lowsrc" />
    <Attribute name="imagePath" />
    <Attribute name="viewClass" />
    <Attribute name="emptyURL" />
    <Attribute name="draftsURL" />
    <Attribute name="folderURL" />
    <Attribute name="prevMonthImage" />
    <Attribute name="nextMonthImage" />
    <Attribute name="style" />
    <Attribute name="content" tag="meta" />
</HTMLRules>
<JSRules>
<!-- Rules for Rewriting JavaScript variables in URLs -->
    <Variable name="URL"> _fr.location </Variable>
    <Variable name="URL"> g_szUserBase </Variable>
    <Variable name="URL"> g_szPublicFolderUrl </Variable>
    <Variable name="URL"> g_szExWebDir </Variable>
    <Variable name="URL"> g_szViewClassURL </Variable>
    <Variable name="URL"> g_szVirtualRoot </Variable>
    <Variable name="URL"> g_szBaseURL </Variable>
    <Variable name="URL"> g_szURL </Variable>
    <Function name="EXPRESSION" name="NavigateTo" paramPatterns="y"/>
</JSRules>
<XMLRules>
    <Attribute name="xmlns"/>
    <Attribute name="href" tag="a"/>
```

```
    <TagText tag="baseroot" />
    <TagText tag="prop2" />
    <TagText tag="prop1" />
    <TagText tag="img" />
    <TagText tag="xsl:attribute"
    attributePatterns="name=src" />
</XMLRules>
</RuleSet>
```

규칙 작성을 위한 절차

다음은 규칙을 작성할 때 따를 수 있는 일반적인 절차입니다 .

- 컨텐트를 다시 작성해야 하는 HTML 페이지가 있는 디렉토리를 확인합니다 .
- 이 디렉토리에서 다시 작성해야 하는 페이지를 확인합니다 .
- 각 페이지에서 다시 작성해야 하는 URL 을 확인합니다 . "http" 및 "/" 를 검색하여 대부분의 URL 이 간단하게 확인할 수 있습니다 .
- URL 의 컨텐트 유형 (HTML, JavaScript 또는 XML) 을 확인합니다 .
- Identity Server 관리 콘솔의 [포털 서버 구성] 아래에 있는 [Rewriter 서비스] 에서 필요한 규칙 집합을 편집하여 이러한 각 URL 을 다시 쓰기 위해 필요한 규칙을 작성합니다 .
- 이러한 모든 규칙을 해당 도메인에 대한 하나의 규칙 집합으로 결합시킵니다 .

규칙 집합 가이드라인

다음 사항에 주의하십시오 .

- 특정 호스트의 우선 순위는 가장 긴 URI 대응부터 결정됩니다 . 예를 들어 , 다음 규칙 집합에서

```
mail1.central.abc.com|iplanet_mail_ruleset
*.sfbay.abc.com|sfbay_ruleset
*.abc.com|generic_ruleset
```

sfbay_ruleset 이 가장 긴 대응이기 때문에 사용됩니다 .

- 규칙 집합의 규칙들은 규칙이 특정 구문과 일치할 때까지 페이지의 각 구문에 차례로 적용됩니다 .

  규칙을 작성할 때 규칙의 순서에 주의하십시오 . 규칙은 규칙 집합에 있는 순서대로 페이지의 구문에 적용됩니다 . 특정한 규칙과 "*" 를 포함한 일반적 규칙이 있는 경우 , 특정한 규칙을 먼저 정의한 다음 일반적 규칙을 적용하십시오 . 그렇게 하지 않으면 특정한 규칙을 발견하기 전에 모든 구문에 일반 규칙이 적용됩니다 .

- 모든 규칙은 `<RuleSet> </RuleSet>` 태그 내에 넣어야 합니다 .
- HTML 컨텐트를 다시 써야 하는 모든 규칙은 규칙 집합의 `<HTMLRules> </HTMLRules>` 부분에 포함시키십시오 .
- JavaScript 컨텐트를 다시 써야 하는 모든 규칙은 규칙 집합의 `<JSRules> </JSRules>` 부분에 포함시키십시오 .
- XML 컨텐트를 다시 써야 하는 모든 규칙은 규칙 집합의 `<XMLRules> </XMLRules>` 부분에 포함시키십시오 .
- 인트라넷 페이지에서 , 다시 써야 하는 URL 을 확인하고 규칙 집합의 해당 부분 (HTML, JSRules 또는 XMLRules) 에 필요한 규칙을 포함시키십시오 .
- 규칙 집합을 필요한 도메인에 할당합니다 . 자세한 내용은 273 페이지의 " 규칙 집합에 대한 URI 의 매핑 목록 만들기 " 를 참조하십시오 .
- 게이트웨이를 다시 시작하여 변경 사항을 적용합니다 .

  gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

규칙 집합의 루트 요소 정의

규칙 집합의 루트 요소에는 두 가지 속성이 있습니다 .

- **`RuleSetName`**. 예를 들면 `default_ruleset` 과 같으며 , 이 이름은 RuleSet 에서 URI 의 매핑으로 참조됩니다 .
- **`Extends`**. 이 속성은 규칙 집합의 상속 기능을 참조합니다 . Extends 값은 규칙 집합을 유도할 기준 집합을 가리킵니다 .

  Extends 값 `none` 을 사용하면 이 새로운 독립 규칙 집합이 다른 규칙 집합에 의존하지 않는 다는 것을 나타내며 *RuleSetName* 을 지정하면 규칙 집합이 다른 규칙 집합에 의존한다는 것을 나타냅니다 .

재귀적 기능 사용

Rewriter 에서는 재귀적 기능을 사용하여 대응되는 문자열 패턴의 끝까지에서 같은 패턴을 검색합니다 .

예를 들어 , Rewriter 에서 다음 문자열을 구문 분석하는 경우 :

```
<a href="src=abc.jpg,src=bcd.jpg,src=xyz.jpg>
```

규칙

```
<Attribute name="href" valuePatterns="*src=**"/>
```

은 처음 나타나는 패턴만을 다시 작성하며 그 결과는 다음과 같습니다 .

```
<a href="src=http://jane.sun.com/abc.jpg,src=bcd.jpg,src=xyz.html>
```

하지만 재귀적 옵션을 사용하면

```
<Attribute name="href" valuePatterns="REC:*src=**"/>;
```

Rewriter 는 대응되는 문자열 패턴에서 끝까지 같은 패턴을 찾기 때문에 다음과 같은 출력을 얻게 됩니다 .

```
<a
href="src=http://jane.sun.com/abc.jpg,src=http://jane.sun.com/bcd.jpg,s
rc=http://jane.sun.com/xyz.jpg>
```

언어 기반 규칙 정의 (규칙 정의)

규칙은 다음 언어를 바탕으로 합니다 .

- HTML
- JavaScript
- XML

HTML 컨텐트에 대한 규칙

웹 페이지의 HTML 컨텐트는 속성 , 폼 및 애플릿으로 더욱 세분할 수 있습니다 . 이에 따라 HTML 컨텐트에 대한 규칙은 다음과 같이 분류됩니다 .

- HTML 컨텐트에 대한 속성 규칙

- HTML 컨텐트에 대한 폼 규칙
- HTML 컨텐트에 대한 애플릿 규칙

HTML 컨텐트에 대한 속성 규칙

이 규칙은 값을 다시 써야 하는 대상 태그의 속성을 확인합니다 . 속성 값은 단순한 URL, JavaScript 또는 DHTML 컨텐트일 수 있습니다 . 예 :

- "img" 태그의 src 속성은 이미지 위치를 가리킵니다 (단순 URL).
- href 속성의 onClick 속성은 링크를 클릭할 때 처리됩니다 (DJS).

이 부분은 다음으로 세분됩니다 .

- 속성 규칙 구문
- 속성 규칙 예제
- DJS 속성 예제

속성 규칙 구문

```
<Attribute name="attributeName" [tag="*" valuePatterns="" source="*"
type="URL|DHTML|DJS"]/>
```

여기서 ,

`attributeName` 은 속성의 이름입니다 (필수).

`tag` 는 속성이 속하는 태그입니다 (옵션 , 기본값 * , 모든 태그를 의미).

`valuePatterns` 105 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오 .

`source` 는 이 속성이 정의되는 페이지의 URI 를 지정합니다 (옵션 , 기본값 * , 모든 페이지를 의미).

`type` 은 값의 유형을 지정합니다 (옵션). 다음이 가능합니다 .

`URL` - 단순 URL(기본값).

`DHMTL` - DHTML 컨텐트 . 이런 종류의 컨텐트는 표준 HTML 컨텐트에서 나타나며 Microsoft 의 HTC 형식 파일에서 사용됩니다 .

`DJS` - JavaScript 컨텐트 . onClick 및 onMouseover 와 같은 모든 HTML 이벤트 처리기는 HTML 속성과 연계된 JavaScript 를 가지고 있습니다 .

속성 규칙 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://mymachine.intranet.com/mypage.html
```

페이지 컨텐트

```
<a href="http://mymachine.intranet.com/mypage.html>
```

규칙

```
<Attribute name="href" />
```

또는

```
<Attribute name="href" tag="a"/>
```

결과

```
<a href=gateway-URL/http://mymachine.intranet.com/myhome.html>
```

설명

다시 작성될 URL 은 이미 절대 URL 이므로 URL 의 접두어로는 게이트웨이 URL 만 사용됩니다 .

DJS 속성 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.sesta.com/focus.html
```

페이지 컨텐트

```
<Form>

<input TYPE=TEXT SIZE=20 value=focus
onClick="Check('/focus.html','focus');return;">

</Form>
```

규칙

```
<Attribute name="onClick" type="DJS"/>

<Function type="URL" name="Check" paramPatterns="y,"/>
```

결과

```
<Form>

<INPUT TYPE=TEXT SIZE=20 value=focus
onClick="Check('gateway-URL/http://abc.sesta.com/focus.html','focus');return
;">
```

```
</Form>
```

설명

지정된 페이지 컨텐트를 다시 쓰기 위해 두 가지 규칙이 필요합니다 . 첫 번째 규칙은 onClick JavaScript 토큰을 확인합니다 . 두 번째 규칙은 다시 작성되어야 하는 check 함수의 매개 변수를 확인합니다 . 이 경우에는 paramPatterns 가 첫 번째 매개 변수 자리에 y 값을 갖기 때문에 첫 번째 매개 변수만 다시 작성됩니다 .

게이트웨이 URL 과 JavaScript 토큰이 나타나는 페이지의 기본 URL 이 필요한 매개 변수 앞에 덧붙여집니다 .

HTML 컨텐트에 대한 폼 규칙

사용자가 찾아보는 HTML 페이지에는 폼이 있을 수 있습니다 . 일부 폼 요소는 URL 을 값으로 취할 수 있습니다 .

이 부분은 다음으로 세분됩니다 .

- 폼 규칙 구문
- 폼 규칙 예제

폼 규칙 구문

```
<Form name="form1" field="visit" [valuePatterns="" source="*"]/>
```

여기서

name 은 폼의 이름입니다 (필수).

field 는 값을 다시 작성해야 하는 폼의 필드입니다 (필수).

valuePatterns 105 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오 .

source 는 이 폼 정의가 있는 html 페이지의 URL 입니다 (옵션 , 기본값 *, 모든 페이지를 의미).

폼 규칙 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://test.siroe.com/testcases/html/form.html
```

페이지 컨텐트

페이지 URI 가 form.html 이고 서버의 루트 디렉토리에 있다고 가정합니다 .

```
<form name=form1 method=POST
action="http://test.siroe.com/testcases/html/form.html">
```

```
<input type=hidden name=abc1 value="0|1234|/test.html">
</form>
```

form1 의 일부인 abc1 이라는 이름의 숨겨진 필드 값에 존재하는 /text.html 을 다시 쓰기 위해 다음 규칙이 필요합니다 .

규칙

```
<Form source="*/form.html" name="form1" field="abc1"
valuePatterns="0|1234|"/>
<Attribute name="action" />
```

결과

```
<FORM name="form1" method="POST"
action="gateway-URL/http://test.siroe.com/testcases/html/form.html">
<input type=hidden name=abc1
value="0|1234|gateway-URL/http://test.siroe.com/test.html">
</FORM>
```

설명

action 태그는 정의된 특정 HTML 속성 규칙을 사용하여 다시 작성됩니다 .

입력 태그 속성 값의 value 는 결과에 나타낸 것처럼 다시 작성됩니다 . 지정된 valuePatterns 를 찾고 일치하는 valuePatterns 이후의 모든 컨텐트는 페이지의 기본 URL 과 게이트웨이 URL 을 앞에 덧붙여 다시 작성됩니다 . 105 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오 .

HTML 컨텐트에 대한 애플릿 규칙

단일 웹 페이지에 많은 애플릿이 있을 수 있고 각 애플릿에는 많은 매개 변수가 있을 수 있습니다 . Rewriter 는 규칙에 지정된 값을 애플릿의 HTML 정의에 대응시키고 애플릿 매개 변수 정의의 일부로 존재하는 URL 값을 수정합니다 . 이러한 교체는 사용자가 특정 웹 페이지를 찾아볼 때가 아니라 서버에서 이루어집니다 . 이 규칙은 HTML 컨텐트의 개체 태그 및 애플릿 모두에서 매개 변수를 찾아 다시 작성합니다 .

이 부분은 다음으로 세분됩니다 .

- 애플릿 규칙 구문
- 애플릿 규칙 예제

애플릿 규칙 구문

<Applet code="*ApplicationClassName/ObjectID*" param="*parametername*" [valuePatterns="" source="*"] />

여기서

code 는 애플릿 또는 개체 클래스의 이름입니다 (필수).

param 은 값을 다시 작성해야 하는 매개 변수의 이름입니다 (필수).

valuePatterns 105 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오 .

source 는 애플릿 정의가 있는 페이지의 URL 입니다 (옵션 , 기본값 *, 모든 페이지를 의미).

애플릿 규칙 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

http://abc.siroe.com/casestudy/test/HTML/applet/rule1.html

페이지 컨텐트

```
<applet codebase="appletcode" code="RewriteURLinApplet.class"
archive="/test.jar">

<param name=Test1 value="/index.html">

</applet>
```

규칙

```
<Applet source="*/rule1.html" code="RewriteURLin*.class" param="Test*"/>
```

결과

```
<APPLET
codebase="gateway-URL/http://abc.siroe.com/casestudy/test/HTML/applet/applet
code" code="RewriteURLinApplet.class" archive="/test.jar">

<param name="Test1" value="gateway-URL/http://abc.siroe.com/index.html">

</APPLET>
```

설명

<Attribute name="codebase"/> 가 default_gateway_ruleset 에서 정의된 규칙이므로 codebase attribute 는 다시 작성됩니다 .

이름이 Test 로 시작되는 모든 매개 변수는 다시 작성됩니다 . 애플릿 코드가 표시되는 페이지의 기본 URL 과 게이트웨이 URL 이 값 param 태그의 value 속성에 접두어로 사용됩니다 .

규칙에서 패턴 매칭 사용

valuePatterns 필드를 사용하여 패턴 매칭을 수행하고 다시 써야 하는 구문의 특정 부분을 확인할 수 있습니다 .

규칙의 일부로 valuePatterns 를 지정했다면 매칭된 패턴 이후의 모든 컨텐트는 다시 작성됩니다 .

아래 예제 폼 규칙을 생각해봅시다 .

```
<Form source="*/source.html" name="form1" field="visit" [valuePatterns="0|1234|"]/>
```

여기서

source 는 폼이 표시되는 html 페이지의 URL 입니다 .

name 은 폼의 이름입니다 .

field 는 값을 다시 써야 하는 폼의 필드입니다 .

valuePatterns 는 다시 써야 하는 문자열 부분을 나타냅니다 . valuePatterns 뒤에 나타나는 모든 컨텐트는 다시 작성됩니다 (옵션 , 기본값 "" 는 전체 값을 다시 써야 함을 나타냄).

valuePatterns 에 특수 문자 지정

특수 문자는 백슬래시로 이스케이프하면 지정할 수 있습니다 . 예 :

```
<Form source="*/source.html" name="form1" field="visit"
[valuePatterns="0|1234|\;original text|changed text"]/>
```

valuePatterns 에서 와일드카드 사용

* 문자를 사용하여 다시 쓰기를 위한 패턴 매칭을 수행할 수 있습니다 .

valuePatterns 필드에 단순히 * 만 지정할 수는 없습니다 . * 는 모든 요소와의 대응을 나타내기 때문에 valuePattern 뒤에 아무 것도 없으며 따라서 Rewriter 가 다시 작성할 것이 없습니다 . * 를 *abc 와 같이 또 다른 문자열과 연결하여 사용할 수 있습니다 . 이 경우에 *abc 이후의 모든 컨텐트가 다시 작성됩니다 .

참고 별표 (*) 는 규칙의 어떤 필드에서나 와일드카드로 사용할 수 있습니다 . 그러나 규칙의 모든 필드에 * 를 포함시킬 수는 없습니다 . 모든 필드에 * 가 있으면 규칙이 무시됩니다 . 오류 메시지는 표시되지 않습니다 .

원본 명령문에서 여러 필드를 구별하기 위해 표시되는 구분 문자 (세미콜론 또는 쉼표) 와 함께 * 또는 ** 를 사용할 수 있습니다 . 하나의 와일드카드 (*) 는 다시 작성되지 않을 모든 필드와 매칭되며 두 개의 와일드카드 (**) 는 다시 작성해야 하는 모든 필드와 매칭됩니다 .

표 3-1 에 * 와일드카드 사용의 몇 가지 실례를 나타내었습니다 .

표 3-1 * 와일드카드 사용의 실례

URL	valuePatterns	설명
url1, url2, url3, url4	valuePatterns = "**, *, **, *"	이 경우에 ** 가 다시 작성해야 하는 부분을 나타내기 때문에 url1 및 url3 이 다시 작성됩니다 .
XYZABChttp://host1.sesta.com/dir1.html	valuePatterns = "*ABC"	이 경우에 http://host1.sesta.com/dir1.html 부분만 다시 작성됩니다 . *ABC 이후 모든 부분을 다시 작성해야 합니다 .
"0|dir1|dir2|dir3|dir4|test|url1	valuePatterns = "*|*|**|*|**|*|"	이 경우에 dir2, dir4 및 url1 이 다시 작성됩니다 . 다시 작성해야 하는 마지막 필드는 ** 를 사용하여 나타내지 않아도 됩니다 .

JavaScript 컨텐트에 대한 규칙

JavaScript 에는 다양한 위치에 URL 이 있을 수 있습니다 . Rewriter 는 JavaScript 를 직접 구문 분석하여 URL 부분을 확인할 수 없습니다 . 특별한 규칙 집합을 작성하여 JavaScript 처리기가 URL 을 확인하여 변환하도록 합니다 .

URL 유형을 가진 JavaScript 요소는 다음으로 분류됩니다 .

- 변수
- 함수의 인수

변수

일반 구문
<Variable name="*variableName*"
[type="URL | EXPRESSION | DHTML | DJS | SYSTEM" source="*"]>

JavaScript 변수는 가지고 있는 값의 유형에 따라 5 가지 범주로 더욱 세분할 수 있습니다 .

- URL 변수
- EXPRESSION 변수
- DHTML(Dynamic HTML) 변수
- DJS(Dynamic JavaScript) 변수
- SYSTEM 변수

URL 변수

변수 값은 URL 로 취급할 수 있는 단순한 문자열입니다 .

이 부분은 다음으로 세분됩니다 .

- URL 변수 구문
- URL 변수 예제

URL 변수 구문

```
<Variable name="variableName" type="URL" [source="*"]>
```

여기서

`variableName` 은 변수의 이름입니다 . `variableName` 의 값은 다시 작성됩니다 (필수).

`type` 은 URL 변수입니다 (필수 , 이 값은 URL 이어야 함).

`source` 는 이 JavaScript 변수가 발견되는 페이지의 URI 입니다 (옵션 , 기본값 * 는 모든 페이지를 의미).

URL 변수 예제

기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.siroe.com/tmp/page.html
```

페이지 컨텐트

```
<script LANGUAGE="Javascript">
<!--
//URL Variables
var imgsrc1="/tmp/tmp.jpg";
var imgsrc2="http://srap.sesta.com/tmp/tmp.jpg";
var imgsrc3=imgsrc2;
//-->
</SCRIPT >
```

규칙

```
<Variable name="imgsrc*" type="URL"/>
```

결과

```
<script LANGUAGE="Javascript">
<!--
//URL Variables
var imgsrc="gateway-URL/http://abc.siroe.com/tmp/tmp.jpg";
var imgsrc="gateway-URL/http://srap.sesta.com/tmp/tmp.jpg";
var imgsrc2=imgsrc1;
//-->
</SCRIPT>
```

설명

유형이 URL 이고 이름이 `imgsrc` 로 시작되는 모든 변수가 다시 작성됩니다 . 결과의 첫 라인에서 게이트웨이 URL 과 변수가 표시되는 페이지의 기본 URL 이 앞에 덧붙습니다 . 두 번째 라인에는 이미 절대 경로가 있기 때문에 게이트웨이 URL 만 덧붙습니다 . 세 번째 var `imagsrc2` 는 그 값이 문자열이 아니라 다른 JavaScript 값이기 때문에 다시 작성되지 않습니다 .

EXPRESSION 변수

Expression 변수에는 오른쪽에 표현식이 있습니다 . 이 표현식의 결과는 URL 입니다 . Rewriter 는 서버에서 이러한 표현식을 평가할 수 없기 때문에 HTML 페이지에 JavaScript 함수 (psSRAPRewriter_convert_expression) 를 추가합니다 . 이 함수는 표현식을 하나의 매개 변수로 받아들여 클라이언트 브라우저에서 필요한 URL 로 평가합니다 .

명령문에 단순 URL 이 있는지 EXPRESSION URL 이 있는지 잘 모르는 경우에는 두 시나리오를 모두 처리할 수 있는 EXPRESSION 규칙을 사용하는 것이 좋습니다 .

이 부분은 다음으로 세분됩니다 .

- EXPRESSION 변수 구문
- EXPRESSION 변수 예제

EXPRESSION 변수 구문

```
<Variable name="variableName" [type="EXPRESSION" source="*"]/>
```

여기서

variableName 은 그 값이 표현식인 JavaScript 변수의 이름입니다 (필수).

type 은 JavaScript 변수의 유형입니다 (옵션 , 기본값은 EXPRESSION).

source 는 페이지의 URI 입니다 (옵션 , 기본값 *, 모든 소스를 의미)

EXPRESSION 변수 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.siroe.com/dir1/dir2/page.html
```

페이지 컨텐트

```
<script LANGUAGE="Javascript">
<!--
//Expression variables
var expvar= getURIPreFix() + "../../images/graphics"+".gif";
document.write("<A HREF="+expvar+">Link to XYZ content</A><P>")
var expvar="../../images/graphics"+".gif";
//-->
</SCRIPT>
```

규칙

```
<Variable name="expvar" type="EXPRESSION"/>
```

또는

```
<Variable name="expvar"/>
```

결과

```
var expvar=psSRAPRewriter_convert_expression(getURIPreFix() +
"../../images/graphics"+".gif");

document.write("<a href="+expvar+">>Link to XYZ content</A><P>")

var expvar="gateway-URL/http://abc.siroe.com/images/graphics"+".gif";
```

설명

첫 번째 라인에서 함수 psSRAPRewriter_convert_expression 이 expvar 표현식 변수의 오른쪽에 덧붙여집니다 . 이 함수는 표현식을 처리하고 런타임에 컨텐트를 다시 작성합니다 . 세 번째 라인에서 값이 단순 URL 로 다시 작성됩니다 .

DHTML(Dynamic HTML) 변수

이것은 HTML 컨텐트를 포함한 JavaScript 변수입니다 .

이 부분은 다음으로 세분됩니다 .

- DHTML 구문
- DHTML 예제

DHTML 구문

```
<Variable name="variableName" type="DHTML" [source="*"]/>
```

여기서

variableName 은 DHTML 컨텐트가 있는 JavaScript 변수의 이름입니다 (필수).

type 은 변수의 유형입니다 (필수 , 이 값은 DHTML 이어야 함).

source 는 페이지의 URL 입니다 (옵션 , 기본값은 *, 모든 페이지를 의미).

DHTML 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.sesta.com/graphics/set1/graphics/jsscript/JSVAR/page.html
```

페이지 컨텐트

```
<script LANGUAGE="Javascript">
<!--
//DHTML Var
var dhtmlVar="<a href=../../images/test.html>"
var dhtmlVar="<a href=/images/test.html>"
var dhtmlVar="<a href=images/test.html>"
//-->
</SCRIPT>
```

규칙

```
<Variable name=”dhtmlVar” type="DHTML"/>
<Attribute name="href" />
```

또는

```
<Attribute name="href" tag="a"/>
```

결과

```
<script LANGUAGE="Javascript">
<!--
//DHTML Var
var dhtmlVar="<a
href=gateway-URL/http://abc.sesta.com/graphics/set1/graphics/images/test.html>"
var dhtmlVar="<a href=gateway-URL/http://abc.sesta.com/images/test.html>"
var dhtmlVar="<a
href=gateway-URL/http://abc.sesta.com/graphics/set1/graphics/jscript/JSVAR/images/test.html>"
//-->
</SCRIPT >
```

설명

JavaScript 구문 분석기는 `dhtmlVar` 의 값을 HTML 컨텐트로 읽고 이 컨텐트를 HTML 구문 분석기로 보냅니다 . HTML 구문 분석기가 href 속성 규칙이 대응된 HTML 규칙을 적용하기 때문에 이 값은 다시 작성됩니다 .

DJS(Dynamic JavaScript) 변수

이것은 JavaScript 컨텐트를 포함한 JavaScript 변수입니다 .

이 부분은 다음으로 세분됩니다 .

- DJS 구문
- DJS 예제

DJS 구문

```
<Variable name="variableName" type="DJS" [source="*"]/>
```

여기서

`variable` 은 그 값이 자바스크립트인 JavaScript 변수입니다 .

DJS 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.sesta.com/dir1/dir2/dir3/jscript/dir4/page.html
```

페이지 컨텐트

```
//DJS Var

var dJSVar="var dJSimgsrc='/tmp/tmp.jpg';"

var dJSVar="var dJSimgsrc='../tmp/tmp.jpg';"

var dJSVar="var dJSimgsrc='http://abc.sesta.com/tmp/tmp.jpg';"
```

규칙

```
<Variable name="DJS">dJSVar/>

<Variable name="URL">dJSimgsrc/>
```

결과

```
//DJS Var - need 2 rules

var dJSVar="var dJSimgsrc='gateway-URL/http://abc.sesta.com/tmp/tmp.jpg';"

var dJSVar="var
dJSimgsrc='gateway-URL/http://abc.sesta.com/dir1/dir2/dir3/jscript/tmp/tmp.jp
g';"

var dJSVar="var dJSimgsrc='gateway-URL/http://abc.sesta.com/tmp/tmp.jpg';"
```

설명

여기에 두 가지 규칙이 필요합니다 . 첫 번째 규칙은 동적 JavaScript 변수 dJSVar 를 찾습니다 . 이 변수의 값은 다시 URL 유형의 JavaScript 입니다 . 이 JavaScript 변수의 값을 다시 쓰기 위해 두 번째 규칙이 적용됩니다 .

SYSTEM 변수

이 변수는 사용자가 선언하는 것이 아니라 JavaScript 표준의 일부로 사용할 수 있는 것입니다 . 예 : window.location.pathname. 이 변수는 제한적으로 지원됩니다 .

이 부분은 다음으로 세분됩니다 .

- SYSTEM 변수 구문
- SYSTEM 변수 예제

SYSTEM 변수 구문

```
<Variable name="variableName" type="SYSTEM" [source="*"]/ >
```

여기서

variableName 은 JavaScript 시스템 변수 (필수) 이며 그 값은 다음 패턴과 매칭되는 값이 될 수 있습니다 . document.URL, document.domain, location, doument.location, location.pathname, location.href, location.protocol, location.hostname, location.host 및 location.port. 이러한 값은 generic_ruleset 에 있습니다 . 이러한 시스템 변수 규칙을 수정하지 마십시오 .

type 은 시스템 유형 값을 지정합니다 (필수이며 값은 DJS).

source 는 이 페이지의 URI 입니다 (옵션 , 기본값은 *, 모든 페이지를 의미).

SYSTEM 변수 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.siroe.com/dir1/page.html
```

페이지 컨텐트

```
<script LANGUAGE="Javascript">
<!--
//SYSTEM Var
alert(window.location.pathname);
//-->
</SCRIPT>
```

규칙

```
<Variable name="window.location.pathname" type="SYSTEM"/ >
```

결과

```
</SCRIPT>

<SCRIPT LANGUAGE="Javascript">

<!--

//SYSTEM Var

alert(psSRAPRewriter_convert_pathname(window.location.pathname));

//-->

</SCRIPT>
```

설명

Rewriter 가 규칙에 대응되는 시스템 변수를 찾으면 `psSRAPRewriter_convert_system` 함수가 접두어로 사용됩니다 . 이 함수는 런타임 때 시스템 변수를 처리하고 그에 따라 얻어지는 URL 을 다시 씁니다 .

함수의 인수

그 값을 다시 작성해야 하는 함수 매개 변수는 4 가지 범주로 분류됩니다 .

- URL 매개 변수
- EXPRESSION 매개 변수
- DHTML 매개 변수
- DJS 매개 변수

일반 구문

```
<Function name="functionName" paramPatterns="y,y,"
[type="URL|EXPRESSION|DHTML|DJS" source="*"]/>
```

여기서

`name` 은 JavaScript 함수의 이름입니다 (필수).

`paramPatterns` 는 다시 작성해야 하는 매개 변수를 지정합니다 (필수).

`y` 의 위치는 다시 작성해야 하는 매개 변수를 나타냅니다 . 예를 들어 구문에서 첫 번째 매개 변수는 다시 써야 하지만 두 번째 매개 변수는 다시 쓰지 않아야 합니다 .

type 은 이 매개 변수에 필요한 값의 종류를 지정합니다 (옵션 , 기본값은 EXPRESSION 유형).

source 페이지 소스 URI(옵션 , 기본값은 *, 모든 페이지를 의미).

URL 매개 변수

함수는 이 매개 변수를 문자열로 취하며 이 문자열은 URL 로 취급할 수 있습니다 .

이 부분은 다음으로 세분됩니다 .

- URL 매개 변수 구문
- URL 매개 변수 예제

URL 매개 변수 구문

```
<Function name="functionName" paramPatterns="y,," type="URL" [source="*"]/>
```

여기서

name 은 URL 유형의 매개 변수를 갖는 함수 이름입니다 (필수).

paramPatterns 는 다시 작성해야 하는 매개 변수를 지정합니다 (필수).

y 의 위치는 다시 작성해야 하는 매개 변수를 나타냅니다 . 예를 들어 구문에서 첫 번째 매개 변수는 다시 써야 하지만 두 번째 매개 변수는 다시 쓰지 않아야 합니다 .

type 은 함수의 유형입니다 (필수 , 이 값은 URL 이어야 함).

source 는 이 함수 호출을 갖는 페이지의 URL 입니다 (옵션 , 기본값은 *, 모든 URL 을 의미).

URL 매개 변수 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.sesta.com/test/rewriter/test1/jscript/test2/page.html
```

페이지 컨텐트

```
<script language="JavaScript">
<!--
function test(one,two,three) {
alert(one + "##" + two + "##" +three);
}
test("/test.html","../test.html","123");
```

```
window.open("/index.html","gen",width=500,height=500);
//-->
</SCRIPT>
```

규칙

```
<Function name="URL" name="test" paramPatterns="y,y,"/>
<Function name="URL" name="window.open" paramPatterns="y,,,"/>
```

결과

```
<SCRIPT language="JavaScript">
<!--
function test(one,two,three) {
alert(one + "##" + two + "##" +three);
}
test("gateway-URL/http://abc.sesta.com/test.html","gateway-URL/http://abc.sesta.com/test/rewriter/test1/jscript/test.html","123");
window.open("gateway-URL/http://abc.sesta.com/index.html","gen",width=500,height=500);
//-->
</SCRIPT>
```

설명

첫 번째 규칙은 `test` 이름의 함수에 있는 처음 두 매개 변수를 다시 작성해야 한다고 지정합니다 . 따라서 test 함수의 처음 두 매개 변수는 다시 작성됩니다 . 두 번째 규칙은 `window.open` 함수의 처음 매개 변수를 다시 작성해야 한다고 지정합니다 . `window.open` 함수 내의 URL 에는 이 함수 매개 변수를 포함한 페이지의 기본 URL 과 게이트웨이 URL 이 앞에 덧붙여집니다 .

EXPRESSION 매개 변수

이 매개 변수는 표현식 값을 취하며 평가 결과는 URL 이 됩니다 .

이 부분은 다음으로 세분됩니다 .

- EXPRESSION 매개 변수 구문
- EXPRESSION 매개 변수 예제

EXPRESSION 매개 변수 구문

```
<Function name="functionName" paramPatterns="y" [type="EXPRESSION"
source="*"]/>
```

여기서

name 은 함수의 이름입니다 (필수).

paramPatterns 는 다시 작성해야 하는 매개 변수를 지정합니다 (필수).

y 의 위치는 다시 작성해야 하는 함수 매개 변수를 나타냅니다 . 위의 구문에서 첫 번째 매개 변수만 다시 작성됩니다 .

type 은 값 EXPRESSION 을 지정합니다 (옵션).

source 는 이 함수가 호출되는 페이지의 URI 입니다 .

EXPRESSION 매개 변수 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.sesta.com/dir1/dir2/page.html
```

페이지 컨텐트

```
<script language="JavaScript">
<!--
function jstest2(){
return ".html";
}
function jstest1(one){
return one;
}
var dir="/images/test"
var test1=jstest1(dir+"/test"+jstest2());
document.write("<a HREF="+test1+">TEST</a>");
alert(test1);
//-->
</SCRIPT>
```

규칙

```
<Function type="EXPRESSION" name="jstest1" paramPatterns="y"/>
```

또는

```
<Function name="jstest1" paramPatterns="y"/>
```

결과

```
<script language="JavaScript">
<!--
function jstest2(){
return ".html";
}
function jstest1(one){
return one;
}
var dir="/images/test"
var
test1=jstest1(psSRAPRewriter_convert_expression(dir+"/test"+jstest2()));
document.write("<a HREF="+test1+">TEST</a>");
alert(test1);
//-->
</SCRIPT>
```

설명

이 규칙은 `jstest1` 함수의 첫 번째 매개 변수를 EXPRESSION 함수 매개 변수로 취급하여 다시 써야 한다는 것을 지정합니다 . 예제 페이지 컨텐트에서 첫 번째 매개 변수는 런타임 때만 평가되는 표현식입니다 . Rewriter 는 이 표현식의 접두어로 `psSRAPRewriter_convert_expression` 함수를 사용합니다 . 이 표현식이 평가되고 `psSRAPRewriter_convert_expression` 함수가 런타임 때 결과를 다시 씁니다 .

참고 위의 예제에서 JavaScript 변수 규칙의 일부로 `test1` 이 있을 필요는 없습니다 . `jstest1` 에 대한 함수 규칙이 다시 쓰기를 처리합니다 .

DHTML 매개 변수

그 값이 HTML 인 함수 매개 변수입니다 .

HTML 페이지를 동적으로 생성하는 `document.write()` 같은 원시 JavaScript 메서드가 이 범주에 속합니다 .

이 부분은 다음으로 세분됩니다 .

- DHTML 매개 변수 구문
- DHTML 매개 변수 예제

DHTML 매개 변수 구문

```
<Function name="functionName" paramPatterns="y" type="DHTML" [source="*"]/>
```

여기서

`name` 은 함수의 이름입니다 .

`paramPatterns` 는 다시 작성해야 하는 매개 변수를 지정합니다 (필수).

`y` 의 위치는 다시 작성해야 하는 함수 매개 변수를 나타냅니다 . 위의 구문에서 첫 번째 매개 변수만 다시 작성됩니다 .

DHTML 매개 변수 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://xyz.siroe.com/test/rewriter/test1/jscript/JSFUNC/page.html
```

페이지 컨텐트

```
<script>
<!--
document.write('<a href="/index.html">write</a><BR>')
document.writeln('<a href="index.html">writeln</a><BR>')
document.write("http://abc.sesta.com/index.html<BR>")
document.writeln("http://abc.sesta.com/index.html<BR>")
//-->
</SCRIPT>
```

규칙

```
<Function name="DHTML" name="document.write" paramPatterns="y"/>
```

```
<Function name="DHTML" name="document.writeln" paramPatterns="y"/>

<Attribute name="href" />
```

결과

```
<SCRIPT>

<!--

document.write('<a
href="gateway-URL/http://xyz.siroe.com/index.html">write</a><BR>')

document.writeln('<a
href="gateway-URL/http://xyz.siroe.com/test/rewriter/test1/jscript/JSFUNC/inde
x.html">writeln</a><BR>')

document.write("http://abc.sesta.com/index.html<BR>")

document.writeln("http://abc.sesta.com/index.html<BR>")

//-->

</SCRIPT>
```

설명

첫 번째 규칙은 `document.write` 함수의 첫 매개 변수를 다시 작성해야 한다고 지정합니다 . 두 번째 규칙은 `document.writeln` 함수의 첫 매개 변수를 다시 작성해야 한다고 지정합니다 . 세 번째 규칙은 `href` 이름의 모든 속성을 다시 작성해야 한다고 지정하는 단순 HTML 규칙입니다 . 이 예에서 , DHTML 매개 변수 규칙이 함수에서 다시 작성해야 하는 매개 변수를 확인합니다 . 그런 다음 HTML 속성 규칙이 적용되어 실제로 확인된 매개 변수가 다시 작성됩니다 .

DJS 매개 변수

그 값이 JavaScript 인 함수 매개 변수입니다 .

이 부분은 다음으로 세분됩니다 .

- DJS 매개 변수 구문
- DJS 매개 변수 예제

DJS 매개 변수 구문

```
<Function name="functionName" paramPatterns="y" type="DJS" [source="*"]/>
```

여기서

`name` 은 하나의 매개 변수가 DJS 인 함수의 이름입니다 (필수).

`paramPatterns` 는 위 함수에서 어떤 매개 변수가 DJS 인지를 지정합니다 (필수).

y 의 위치는 다시 작성해야 하는 함수 매개 변수를 나타냅니다 . 위의 구문에서 첫 번째 매개 변수만 다시 작성됩니다 .

type 은 DJS 입니다 (필수).

source 는 페이지의 URI 입니다 (옵션 , 기본값은 *, 모든 URI 를 의미).

DJS 매개 변수 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.sesta.com/page.html
```

페이지 컨텐트

```
<script>

menu.addItem(new NavBarMenuItem("All Available
Information","JavaScript:top.location='http://abc.sesta.com'"));

</script>
```

규칙

```
<Function name="DJS" name="NavBarMenuItem" paramPatterns=",y"/>

<Variable name="URL">top.location</Variable>
```

결과

```
<script>

menu.addItem(new NavBarMenuItem("All Available
Information","JavaScript:top.location='gateway-URL/http://abc.sesta.com'"));

</script>
```

설명

첫 번째 규칙은 JavaScript 를 포함한 NavBarMenuItem 함수의 두 번째 매개 변수를 다시 작성해야 한다고 지정합니다 . JavaScript 내에서 , top.location 변수도 다시 작성해야 합니다 . 이 변수는 두 번째 규칙을 사용하여 다시 작성됩니다 .

XML 컨텐트에 대한 규칙

웹 페이지에 XML 컨텐트가 있을 수 있으며 여기에는 다시 URL 이 있을 수 있습니다 . 다시 작성해야 하는 XML 컨텐트는 두 범주로 구분됩니다 .

- 태그 텍스트 (태그의 PCDATA 또는 CDATA 와 동일)

- 속성

태그 텍스트

이 규칙은 태그 요소의 PCDATA 또는 CDATA 를 다시 작성하기 위한 것입니다 .

이 부분은 다음으로 세분됩니다 .

- 태그 텍스트 구문
- 태그 텍스트 예제

태그 텍스트 구문

```
<TagText tag="tagName" [attributePatterns="attribute_patterns_for_
this_tag" source="*"]/>
```

여기서

tagName 은 태그의 이름입니다 .

attributePatterns 는 이 태그에 대한 속성 및 그 값의 패턴입니다 (옵션 , 이 태그에 속성이 전혀 없다는 것을 의미).

source 는 이 xml 파일의 URI 입니다 (옵션 , 기본값은 *, 모든 xml 페이지를 의미).

태그 텍스트 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다 .

```
http://abc.sesta.com/test/rewriter/test1/xml/page.html
```

페이지 컨텐트

```
<xml>

<Attribute name="src">test.html</attribute>

<attribute>abc.html</attribute>

</xml>
```

규칙

```
<TagText tag="attribute" attributePatterns="name=src"/ >
```

결과

```
<xml>

<Attribute
name="src">gateway-URL/http://abc.sesta.com/test/rewriter/test1/xml/test.html<
/attribute>
```

```
<attribute>abc.html</attribute>

</xml>
```

설명

페이지 컨텐트의 첫 번째 행에는 속성 예제가 있습니다. 페이지 컨텐트의 두 번째 행에 name 이라는 속성이 없는데 name 속성의 값이 `src` 여야 하기 때문에 다시 작성하는 작업은 수행되지 않습니다. 이것도 다시 작성하려면 `<TagText tag="attribute"/>` 가 있어야 합니다.

속성

XML 속성의 규칙은 HTML 에 대한 속성 규칙과 유사합니다. 118 페이지의 "HTML 컨텐트에 대한 속성 규칙 " 을 참조하십시오. XML 의 속성 규칙이 대소문자를 구분하는 반면 HTML 속성은 그렇지 않다는 차이점이 있습니다. 이는 근본적으로 HTML 에는 없지만 XML 에는 있는 대소문자 구분 특성 때문입니다.

Rewriter 는 속성 이름을 바탕으로 속성 값을 변환합니다.

이 부분은 다음으로 세분됩니다.

- 속성 구문
- 속성 예제

속성 구문

<Attribute name="*attributeName*" [tag="*" type="URL" valuePatterns="*" source="*"]/>

여기서

`attributeName` 은 속성의 이름입니다 (필수).

`tag` 는 이 속성이 있는 태그의 이름입니다 (옵션 , 기본값은 *, 모든 태그를 의미).

`valuePatterns` 105 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오.

`source` 는 이 XML 페이지의 URI 입니다 (옵션 , 기본값은 *, 모든 XML 페이지를 의미).

속성 예제

페이지의 기본 URL 이 다음과 같다고 가정합니다.

```
http://abc.sesta.com/test/rewriter/test1/xml/page.html
```

페이지 컨텐트

```
<xml>
<baseroot href="/root.html"/>
<img href="image.html"/>
<string href="1234|substring.html"/>
<check href="1234|string.html"/>
</xml>
```

규칙

```
<Attribute name="href"tag="check" valuePatterns="1234|"/>
```

결과

```
<xml>
<baseroot href="/root.html"/>
<img href="image.html"/>
<string href="1234|substring.html"/>
<check
href="1234|gateway-URL/http://abc.sesta.com/test/rewriter/test1/xml/string.html"/>
</xml>
```

설명

위의 예에서 네 번째 라인만 규칙에 지정된 모든 조건을 만족하기 때문에 이 라인만 다시 작성됩니다 . 116 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오 .

CCS(Cascading Style Sheet) 에 대한 규칙

HTML 페이지에서 CCS(CCS2 포함) 가 변환됩니다 . URL 이 CSS 의 가져오기 구문과 `url()` 함수에만 있기 때문에 이 변환에 대해 정의된 규칙은 없습니다 .

WML 에 대한 규칙

WML 은 HTML 과 유사하며 따라서 WML 컨텐트에는 HTML 규칙이 적용됩니다 . WML 컨텐트에 일반 규칙 집합을 사용하십시오 . 99 페이지의 "HTML 컨텐트에 대한 규칙 " 을 참조하십시오 .

재귀적 기능 사용

Rewriter 에서는 재귀적 기능을 사용하여 대응되는 문자열 패턴의 끝까지에서 같은 패턴을 검색합니다 .

예를 들어 , Rewriter 에서 다음 문자열을 구문 분석하는 경우 :

```
<a href="src=abc.jpg,src=bcd.jpg,src=xyz.jpg>
```

규칙

```
<Attribute name="href" valuePatterns="*src=**"/>
```

은 처음 나타나는 패턴만을 다시 작성하며 그 결과는 다음과 같습니다 .

```
<a href="src=http://jane.sun.com/abc.jpg,src=bcd.jpg,src=xyz.html>
```

하지만 재귀적 옵션을 사용하면

```
<Attribute name="href" valuePatterns="REC:*src=**"/>;
```

Rewriter 는 대응되는 문자열 패턴에서 끝까지 같은 패턴을 찾기 때문에 다음과 같은 출력을 얻게 됩니다 .

```
<a
href="src=http://jane.sun.com/abc.jpg,src=http://jane.sun.com/bcd.jpg,s
rc=http://jane.sun.com/xyz.jpg>
```

게이트웨이 서비스에서 Rewriter 구성

[Rewriter] 탭 아래에서 게이트웨이 서비스를 사용하여 기본 및 고급의 두 범주 내에서 다음 작업을 수행할 수 있습니다 .

- 기본 작업
  - 모든 URL 다시 쓰기 사용
  - RuleSet 과 URI 의 매핑 목록 만들기

- 구문 분석할 MIME 유형 목록 만들기
- 기본 도메인 지정

• 고급 작업

- 다시 쓰지 않을 URI 목록 만들기
- MIME 추측 사용
- 구문 분석할 URI 매핑 목록 만들기
- 마스크 활성화
- 마스크 씨드 문자열 지정
- 마스크하지 않을 URI 목록 만들기
- 게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 표시하기

기본 작업

모든 URL 다시 쓰기 사용

게이트웨이 서비스에서 [모든 URL 다시 쓰기 사용] 옵션을 설정하면 [도메인 및 부속 도메인의 프록시] 목록에 있는 항목을 확인하지 않고 Rewriter 가 모든 URL 을 다시 씁니다 . [도메인 및 부속 도메인의 프록시] 목록에 있는 항목은 무시됩니다 .

➤ **게이트웨이가 모든 URL 을 다시 쓰도록 하려면**

1. Sun Java™ System Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 대한 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [Rewriter] 탭을 클릭합니다 .
6. [모든 URL 다시 쓰기 사용] 확인란을 선택하여 게이트웨이가 모든 URL 을 다시 쓸 수 있게 합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

RuleSet 과 URI 의 매핑 목록 만들기

규칙 집합은 Identity Server 관리 콘솔의 포털 서버 구성 아래의 Rewriter 서비스에서 만들어집니다 . 자세한 내용은 ***Portal Server** 관리 설명서*를 참조하십시오 .

규칙 집합을 만든 후 [규칙 집합에 URI 매핑] 필드를 사용하여 도메인을 규칙 집합과 연관시킵니다 . 기본적으로 다음 두 항목이 [규칙 집합에 URI 매핑] 필드에 추가됩니다 .

- `*://*.Sun.COM/portal/*|default_gateway_ruleset`

  여기서 `sun.com` 은 포털의 설치 도메인이고 /portal 은 포털 설치 컨텍스트입니다 .

- `*|generic_ruleset`

즉 , 도메인 `sun.com` 을 가진 포털 디렉토리의 모든 페이지에 `default_gateway_ruleset` 이 적용됩니다 . 기타 모든 페이지에는 일반 규칙 집합이 적용됩니다 . `default_gateway_ruleset` 및 `generic_ruleset` 은 사전 구성된 규칙 집합입니다 .

참고 표준 포털 데스크탑에 나타나는 모든 컨텐트에 대해 컨텐트를 가져온 위치에 관계 없이 `default_gateway_ruleset` 의 규칙 집합이 사용됩니다 .

예를 들어 , 표준 포털 데스크탑이 URL `yahoo.com` 에서 컨텐트를 스크랩하도록 구성되었다고 가정합시다 . Portal Server 는 `sesta.com` 에 있습니다 . `sesta.com` 에 대한 규칙 집합이 불려온 컨텐트에 적용됩니다 .

참고 규칙 집합을 지정하는 도메인은 [도메인 및 부속 도메인의 프록시] 목록에 있어야 합니다 .

구문 내에서 와일드카드 사용

규칙 집합에서 별표를 사용하여 완전한 정규 URI 또는 부분적 URI 를 매핑할 수 있습니다 .

예를 들어 , 다음과 같이 java_index_page_ruleset 을 index.html 페이지에 적용할 수 있습니다 .

```
www.sun.com/java/index.html/java_index_page_ruleset
```

또는 다음과 같이 java 디렉토리에 있는 모든 페이지를 java_directory_ruleset 에 적용할 수 있습니다 .

```
www.sun.com/java/* /java_directory_ruleset
```

➤ URI 를 규칙 집합에 매핑하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭을 클릭합니다 .
6. [규칙 집합에 URI 매핑] 필드로 스크롤합니다 .
7. 필요한 도메인이나 호스트 이름 그리고 규칙 집합을 [규칙 집합에 URI 매핑] 필드에 입력하고 [추가] 를 누릅니다 .

   항목이 [규칙 집합에 URI 매핑] 필드에 추가됩니다 .

   도메인이나 호스트 이름 그리고 규칙 집합을 지정하는 형식은 다음과 같습니다 .

   ```
   domain name|ruleset name
   ```

   예 :

   ```
   eng.sesta.com|default
   ```

구문 분석할 MIME 유형 목록 만들기

Rewriter 에는 컨텐트 유형 (HTML, JAVASCRIPT, CSS 및 XML) 에 따라 웹 페이지의 구문 분석에 사용되는 4 가지 구문 분석기가 있습니다 . 기본적으로 공통 MIME 유형이 이러한 구문 분석기와 연결되어 있습니다 . 게이트웨이 서비스의 [구문 분석기를 MIME 유형에 매핑] 필드에서 새로운 MIME 유형을 이러한 구문 분석기와 연관시킬 수 있습니다 . 그러면 Rewriter 의 기능이 다른 MIME 유형까지 확장됩니다 .

여러 항목을 입력할 때는 세미콜론이나 콤마로 구분합니다 (";" 또는 ","). 예 :

```
HTML=text/html;text/htm;text/x-component;text/wml; text/vnl/wap.wml
```

이것은 이러한 MIME 을 가진 모든 컨텐트가 HTML Rewriter 로 보내지고 URL 을 다시 쓰도록 HTML 규칙이 적용된다는 의미입니다 .

팁 MIME 매핑에서 불필요한 구문 분석기를 제거하면 작동 속도를 높일 수 있습니다 . 예를 들어 , 특정 인트라넷의 컨텐트에 JavaScript 가 없다는 것이 확실하면 MIME 매핑 목록에서 JAVASCRIPT 항목을 제거할 수 있습니다 .

➤ MIME 매핑을 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭을 클릭합니다 .
6. [구문 분석기를 MIME 유형에 매핑] 필드로 스크롤하여 편집 상자에 필요한 MIME 유형을 추가합니다 . 여러 항목을 구분할 때는 세미콜론이나 콤마를 사용합니다 .

   항목을 `HTML=text/html;text/htm` 형식으로 지정합니다 .
7. [추가] 를 눌러 목록에 필요한 항목을 추가합니다 .
8. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
9. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   portal-server-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

다시 쓰지 않을 URI 목록 만들기

➤ 다시 쓰지 않을 URI 를 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [기본] 부분을 클릭합니다 .
6. [다시 쓰지 않을 URI] 필드로 스크롤하여 편집 상자에 URI 를 추가합니다 .

   참고 : 이 목록에 #* 를 추가하면 href 규칙이 규칙 집합의 일부라 하더라도 URI 를 다시 쓸 수 있습니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

기본 도메인 지정

기본 도메인 및 부속 도메인은 URL 에 도메인과 부속 도메인 없이 호스트 이름만 있을 때 유용합니다 . 이 경우에 게이트웨이는 호스트 이름이 기본 도메인 및 부속 도메인에 있다고 가정하고 그에 따라 진행합니다 .

예를 들어 , URL 의 호스트 이름이 host1 이고 기본 도메인과 부속 도메인이 red.sesta.com 으로 지정된 경우 , 호스트 이름은 host1.red.sesta.com 으로 확인됩니다 .

➤ 기본 도메인을 지정하려면

1. Identity 서버 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 클릭합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 오른쪽 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 대한 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [기본 도메인] 필드로 스크롤하여 subdomain.domain name 형식으로 필요한 기본값을 입력합니다 .

6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   portal-server-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

고급 작업

MIME 추측 사용

Rewriter 는 페이지의 MIME 유형에 따라 구문 분석기를 선택합니다 . WebLogic 및 Oracle 같은 일부 웹 서버는 MIME 유형을 보내지 않습니다 . 이 문제를 해결하려면 [구문 분석기와 URI 의 매핑] 목록 상자에 데이터를 추가하여 MIME 추측 기능을 활성화합니다 .

➤ **MIME 추측을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [MIME 추측 사용] 확인란을 선택하여 MIME 추측의 사용을 설정합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

구문 분석할 URI 매핑 목록 만들기

[MIME 추측] 확인란이 선택되어 있고 서버에서 MIME 유형을 보내지 않은 경우에는 이 목록 상자를 사용하여 구문 분석할 URI 를 매핑합니다 .

각 URI 는 세미콜론으로 구분합니다 .

예 : HTML=*.html; *.htm;*Servlet

이 것은 html, htm 또는 Servlet 확장을 가진 모든 페이지에 대한 컨텐트를 다시 쓰기 위해 HTML Rewriter 가 사용된다는 것을 의미합니다 .

➤ URI 매핑을 구문 분석하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [구문 분석기를 MIME 유형에 매핑] 필드로 스크롤하여 편집 상자에 데이터를 추가합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

마스크 활성화

마스크를 사용하면 Rewriter 에서 페이지의 인트라넷 URL 이 보이지 않도록 URI 를 다시 쓸 수 있습니다 .

➤ 마스크를 활성화하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [마스크 사용] 확인란을 선택하여 마스크를 활성화합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

마스크 씨드 문자열 지정

씨드 문자열은 URI 의 마스크에 사용됩니다 . 씨드 문자열은 마스크 알고리즘에 의해 생성되는 임의 문자열입니다 .

참고 이 씨드 문자열을 변경하거나 게이트웨이를 다시 시작한 경우에는 마스크된 URI 의 책갈피가 제대로 작동하지 않을 수 있습니다 .

➤ **마스크 씨드 문자열을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [마스크 씨드 문자열] 필드로 스크롤하고 편집 상자에 문자열을 추가합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

마스크하지 않을 URI 목록 만들기

일부 응용 프로그램 (애플릿 등) 은 인터넷 URI 가 필요하기 때문에 마스크할 수 없습니다 . 이러한 응용프로그램을 지정하려면 URI 를 목록 상자에 추가합니다 .

예를 들어 다음을 목록 상자에 추가하면

```
*/Applet/Param*
```

컨텐트 URI `http://abc.com/Applet/Param1.html` 이 규칙 집합의 규칙에서 매칭되는 경우 URL 이 마스크되지 않습니다 .

➤ 마스크하지 않을 URI 목록을 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭의 [고급] 부분을 누릅니다 .
6. [마스크하지 않을 URI] 필드로 스크롤하여 편집 상자에 URI 를 추가합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 표시하기

게이트웨이가 HTTP 와 HTTPS 모드 모두에서 실행되는 경우 Rewriter 가 일관된 프로토콜을 사용하여 HTML 컨텐트의 참조 자원에 액세스하게 할 수 있습니다 .

예를 들어 원본 URL 이 `http://intranet.com/Public.html` 이라면 HTTP 게이트웨이가 추가됩니다 . 원본 URL 이 `https://intranet.com/Public.html` 이라면 HTTPS 게이트웨이가 추가됩니다 .

참고 이는 Javascript 로 생성된 동적 URI 가 아닌 정적 URI 에만 적용됩니다 .

➤ 게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .

6. [게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 표시] 확인란을 선택합니다.
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

디버깅 로그 사용의 문제 해결

Rewriter 문제를 해결하려면 디버깅 로그를 사용해야 합니다 .

디버깅 메시지는 다음과 같이 분류됩니다 .

- **error**–Rewriter 가 복구할 수 없는 오류입니다 .
- **warning** –이 파일에는 경고 메시지에 대한 로그가 들어 있습니다 . Rewriter 는 이 유형의 오류를 복구할 수 있지만 약간의 오작동이 생길 수도 있습니다 . 예를 들어 경고 메시지로 " 이미지 컨텐트 다시 쓰지 않음 " 이 기록될 수 있습니다 . Rewriter 는 원래 이미지를 다시 쓰지 않기 때문에 이것은 문제라기보다 단지 경고일 뿐이고 Rewriter 의 기능에 심각한 영향을 미치지는 않습니다 . 경고 메시지 일부는 정보 제공을 위한 것입니다 .
- **message**–Rewriter 가 제공하는 가장 높은 수준의 정보입니다 .

Rewriter 디버깅 수준 설정

➤ **Rewriter 디버깅 수준을 설정하려면**

1. 게이트웨이 컴퓨터에 루트로 로그인하여 다음 파일을 편집합니다 .

   gateway-install-root/SUNWam/config/AMConfig-*instance-name*.properties

2. 디버깅 수준을 설정합니다 .

   com.iplanet.services.debug.level=

   디버깅 수준은 다음과 같습니다 .

   error - 심각한 오류만 디버깅 파일에 기록됩니다 . 이런 오류가 발생하면 보통 Rewriter 가 중지됩니다 .

   warning - 경고 메시지가 기록됩니다 .

   message - 모든 디버깅 메시지가 기록됩니다 .

   off - 디버깅 메시지가 기록되지 않습니다 .

3. AMConfig-*instance-name*.properties 파일의 다음 속성에 디버그 파일의 디렉토리를 지정합니다 .

   com.iplanet.services.debug.directory=/var/opt/SUNWam/debug

   여기서 /var/opt/SUNWam/debug 는 기본 디버깅 디렉토리입니다 .

4. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

디버깅 파일 이름

디버깅 수준이 메시지로 설정되면 디버깅이 일단의 파일을 생성합니다 . 표 3-2 에 Rewriter 파일과 여기에 포함된 정보를 나타내었습니다 .

표 3-2 Rewriter 디버깅 파일

파일 이름	정보
RuleSetInfo	재작성에 사용된 모든 규칙 집합이 이 파일에 기록됩니다 .
Original Pages	페이지 URI, resolveURI(페이지 URI 와 다른 경우), 컨텐트 MIME, 페이지에 적용된 규칙 집합 , 구문 분석기 MIMIE 및 원본 컨텐트가 들어 있습니다 . 구문 분석과 관련된 특정 오류 / 경고 / 메시지도 이 파일에 들어 있습니다 . 메시지 모드에서는 전체 컨텐트가 기록되고 경고와 오류 모드에서는 재작성 중에 발생한 예외만 기록됩니다 .
Rewritten Pages	페이지 URI, resolveURI(페이지 URI 와 다른 경우), 컨텐트 MIME, 페이지에 적용된 규칙 집합 , 구문 분석기 MIMIE 및 재작성된 컨텐트가 들어 있습니다 . 이 파일은 디버그 모드가 메시지로 설정되었을 때 채워집니다 .
Unaffected Pages	수정되지 않은 페이지 목록을 포함합니다 .
URIInfo Pages	이 파일에는 발견되어 변환된 URL 이 들어 있습니다 . 컨텐트가 원본 데이터와 동일하게 유지되는 모든 페이지의 세부 사항이 이 파일에 기록됩니다 . 세부적으로 기록되는 내용 : 페이지 URI, MIME 및 인코딩 데이터 , 재작성에 사용된 rulesetID 그리고 구문 분석기 MIME.

위의 파일 이외에도 Rewriter 는 위 파일에서 포착하지 않은 디버그 메시지의 파일을 생성합니다 . 이 파일 이름은 두 부분으로 구성됩니다 . 첫 번째 부분은 pwRewriter 또는 psSRARewriter 이고 두 번째 부분은 portal 또는 *gateway-profile-name* 을 사용한 확장자입니다 .

디버깅 파일은 포털 또는 게이트웨이에 표시됩니다 . 이 파일은 AMConfig-*instance-name*.properties 파일에 표시된 디렉토리에 있습니다 .

Rewriter 구성 요소는 다음 파일 집합을 생성하여 디버깅을 지원합니다 .

*prefix*_RuleSetInfo.*extension*

*prefix*_OrginalPages.*extension*

*prefix*_RewrittenPages.*extension*

*prefix*_UnaffectedPages.*extension*

*prefix*_URIInfo.*extension*

여기서

prefix 는 URLScraper 사용 로그의 경우 `psRewriter` 이거나 게이트웨이 사용 로그의 경우 `psSRAPRewriter` 입니다 .

extension 은 URLScaper 사용의 경우 `portal`, 게이트웨이 사용의 경우 *`gateway-profile-name`* 입니다 .

예를 들어 , 게이트웨이에서 Rewriter 가 페이지를 변환하는데 사용되고 기본 게이트웨이 프로필이 사용되는 경우 디버깅이 다음 파일을 만듭니다 .

```
psSRAPRewriter_RuleSetInfo.default
psSRAPRewriter_OriginalPages.default
psSRAPRewriter_RewrittenPages.default
psSRAPRewriter_UnaffectedPages.default
psSRAPRewriter_URIInfo.default
psSRAPRewriter.default
```

작업 예제

이 절은 다음 내용으로 이루어져 있습니다 .

- 다시 작성해야 하는 컨텐트를 가진 단순 HTML 페이지
- 컨텐트를 다시 작성할 때 필요한 규칙
- 다시 작성된 해당 HTML 페이지

이러한 예제 페이지는 *portal-server-URL*/rewriter 디렉토리에 있습니다. 규칙을 적용하기 전에 페이지를 둘러본 다음 게이트웨이를 통해 재작성된 결과 파일을 검토하여 규칙의 작용에 대해 알아봅니다. 규칙이 이미 default_gateway_ruleset 의 일부인 경우도 있습니다. 어떤 예제에서는 default_gateway_ruleset 에 규칙을 포함시켜야 할 수 있습니다. 해당 위치에서 이에 대해 언급합니다.

참고 굵게 표시된 부분은 다시 작성된 내용입니다.

다음 예제를 이용할 수 있습니다.

- HTML
  - HTML 속성 예제
  - HTML 폼 예제
  - HTML 애플릿 예제
- JavaScript
  - 변수
    - JavaScript URL 변수 예제
    - JavaScript 컨텐트 예제
    - JavaScript DHTML 변수 예제
    - JavaScript DJS 변수 예제
    - JavaScript SYSTEM 변수 예제
  - 함수
    - JavaScript URL 함수 예제
    - JavaScript EXPRESSION 함수 예제
    - JavaScript DHTML 함수 예제
    - JavaScript DJS 함수 예제
- XML
  - XML 속성 예제

HTML 컨텐트 예제

HTML 속성 예제

➤ **HTML 속성 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/HTML/attrib/attribrule.html

2. 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 `abc.sesta.com` 과 `host1.siroe.com` 이 정의되어 있어야 합니다 .

   정의되어 있지 않으면 직접 연결이 가정되고 게이트웨이 URL 이 앞에 덧붙지 않습니다 .

이 샘플에서 지정한 규칙은 이미 정의되어 있기 때문에 `default_gateway_ruleset` 에 추가할 필요가 없습니다 .

재작성 전의 HTML

```
<html>

Rewriting starts

<head>

<title>TEST PAGE () </title>

</head>

ID-htmlattr.1

<br><br>

1.a href <a href="http://abc.sesta.com/images/logo.gif">http://..</a>

<br><br>

2. href <a href="https://host1.siroe.com">https://..</a>

<br><br>

3. href <a href="../images/logo.gif">../images/</a>

<br><br>

4. href <a href="images/logo.gif">images/..</a> <br><br>

5. href <a href="../../images/logo.gif">../../images/</a> <br><br>

Rewriting ends

</html>
```

규칙

```
<Attribute name="href" />
```

재작성 후의 HTML

```
<html >

Rewriting starts

<head>

<title>TEST PAGE () </title>

</head>

ID-htmlattr.1

<br><br>
```

**1. a href http://..
**

// <Attrib name="href"/> 규칙이 default_gateway_ruleset 에 이미 정의되었기 때문에 이 URL 은 다시 작성됩니다 . URL 이 이미 절대 경로이므로 게이트웨이 URL 만 접두어로 사용됩니다 . 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 abc.sesta.com 이 정의되어 있어야 합니다 . 그렇게 하지 않으면 직접 연결이 가정되기 때문에 게이트웨이 URL 이 접두어로 사용되지 않습니다 .

2. href https://..

// 다시 한번 , 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 host1.siroe.com 이 정의되어 있어야 합니다 . 그렇게 하지 않으면 직접 연결이 가정되기 때문에 게이트웨이 URL 이 접두어로 사용되지 않습니다 .

```
<br><br>
```

3. href ../images/

// 상대 경로가 지정되었기 때문에 필요한 하위 디렉토리와 함께 게이트웨이 URL 과 Portal Server URL 이 접두어로 사용됩니다 . 제공된 샘플 구조에서 HTML 디렉토리 아래에 images 라는 디렉토리가 없기 때문에 이 링크가 작동하지 않습니다 .

```
<br><br>
```

**4 href images/..

**

// 상대 경로가 지정되었기 때문에 필요한 하위 디렉토리와 함께 게이트웨이 URL 과 Portal Server URL 이 접두어로 사용됩니다 .

**5. href ../../images/

**

// 상대 경로가 지정되었기 때문에 필요한 하위 디렉토리와 함께 게이트웨이 URL 과 Portal Server URL 이 접두어로 사용됩니다 . 제공된 샘플 구조에서 `Rewriter` 디렉토리 아래에 `images` 라는 디렉토리가 없기 때문에 이 링크가 작동하지 않습니다 .

```
Rewriting ends
</html>
```

HTML 동적 JavaScript 토큰 예제

➤ **HTML JavaScript 토큰 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/HTML/jstokens/JStokens.html

2. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 `default_gateway_ruleset` 에 추가하십시오 .

3. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 `default_gateway_ruleset` 을 편집합니다 .

4. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

재작성 전의 HTML

```
<html>
<head>
Rewriting starts
<script language="javascript">
function Check(test,ind){
if (ind == 'blur')
{alert("testing onBlur")}
```

```
if (ind == 'focus')
{alert("testing onFocus")}
}
</SCRIPT>
</head>
<body>
<form>
<input TYPE=TEXT SIZE=20 value=blur
onAbort="Check('/indexblur.html','blur');return;">
<input TYPE=TEXT SIZE=20 value=blur
onBlur="Check('/indexblur.html','blur');return;">
<input TYPE=TEXT SIZE=20 value=focus
onFocus="Check('/focus.html','focus');return;">
<input TYPE=TEXT SIZE=20 value=focus
onChange="Check('/focus.html','focus');return;">
<input TYPE=TEXT SIZE=20 value=focus
onClick="Check('/focus.html','blur');return;">
<br><br>
</form>
</body>
Rewriting ends
</html>
```

규칙

```
<Attribute name="onClick" type="DJS"/>
<Function type="URL" name="Check" paramPatterns="y"/>
```

참고 `<Function name="URL" name="Check" paramPatterns="y"/>` 는 JavaScript 함수 규칙이며 JavaScript 함수 예제에 자세하게 설명되어 있습니다 .

재작성 후의 HTML

```
<html>
<head>
```

```
Rewriting starts

<script language="javascript">

function Check(test,ind){

if (ind == 'blur')

{alert("testing onBlur")}

if (ind == 'focus')

{alert("testing onFocus")}

}

</SCRIPT>

</head>

<body>

<form>

<input TYPE=TEXT SIZE=20 value=blur onAbort="Check('gateway
URL/portal-server-URL/indexblur.html','blur');return;">

<input TYPE=TEXT SIZE=20 value=blur onBlur="Check('gateway
URL/portal-server-URL/indexblur.html','blur');return;">

<input TYPE=TEXT SIZE=20 value=focus onFocus="Check('gateway
URL/portal-server-URL/focus.html','focus');return;">

<input TYPE=TEXT SIZE=20 value=focus onChange="Check('gateway
URL/portal-server-URL/focus.html','focus');return;">

<input TYPE=TEXT SIZE=20 value=focus onClick="Check('gateway
URL/portal-server-URL/focus.html','blur');return;">
```

// 이 샘플에서는 모든 명령문이 다시 작성됩니다 . 각 경우에 게이트웨이 및 Portal Server URL 이 접두어로 사용됩니다 . onAbort, onBlur, onFocus, onChange 및 onClick 에 대한 규칙이 default_gateway_ruleset 파일에 정의되었기 때문입니다 . Rewriter 는 JavaScript 토큰을 검색한 다음 추가 처리를 할 수 있도록 JavaScript 함수 규칙으로 전달합니다 . 샘플의 두 번째 규칙은 Rewriter 에 다시 작성할 매개 변수를 알려줍니다 .

```
</body>

<br>

Rewriting ends

</html>
```

HTML 폼 예제

➤ 폼 예제를 사용하려면

1. 다음 위치에서 예제에 액세스합니다 .

   portal-server-URL/rewriter/HTML/forms/formrule.html

2. 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 `abc.sesta.com` 이 정의되어 있어야 합니다 .

   정의되어 있지 않으면 직접 연결이 가정되고 게이트웨이 URL 이 앞에 덧붙지 않습니다 .

3. 이 예제에 지정된 규칙을 "HTML 속성 재작성을 위한 규칙 " 부분의 `default_gateway_ruleset` 에 추가하십시오 .

4. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 `default_gateway_ruleset` 을 편집합니다 .

5. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

재작성 전의 HTML 페이지

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html>

<head>

</head>

<body>

RW_START

<p>

<form name="form1" method="Post"
action="http://abc.sesta.com/casestudy/html/form.html">

<input type="hidden" name="name1" value="0|1234|/test.html">

<input type="hidden" name="name3" value="../../html/test.html">

<form name="form2" method="Post"
action="http://abc.sesta.com/testcases/html/form.html"><br>

<input type="hidden" name="name1"
value="0|1234|../../html/test.html"></form>

RW_END </p>
```

```
</body>

</html>
```

규칙

```
<Form source="*" name="form1" field="name1" valuePatterns="0|1234|"/>
```

재작성 후의 HTML 페이지

```
<HTML>

<HEAD>

RW_START

</HEAD>

<BODY>

<P>

<FORM name=form1 method=POST
action="gateway-URL/http://abc.sesta.com/casestudy/html/form.html">
```

// `<Attribute name="action"/>` 이 `default_gateway_ruleset` 에서 HTML 규칙의 일부로 정의되었기 때문에 이 URL 은 다시 작성됩니다 . URL 이 이미 절대 경로이므로 접두어로 게이트웨이 URL 만 사용하면 됩니다 . 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 `abc.sesta.com` 이 정의되어 있어야 합니다 . 그렇게 하지 않으면 직접 연결이 가정되기 때문에 게이트웨이 URL이 접두어로 사용되지 않습니다 .

```
<input type=hidden name=name1 value="0|1234|gateway
URL/portal-server-URL/test.html">
```

// 여기서 폼 이름은 `form1` 이고 필드 이름은 `name1` 입니다 . 이것은 규칙에서 지정된 폼 이름 및 필드 이름과 일치합니다 . 규칙에 `valuePatterns` 가 `0|1234|` 로 나와있으며 여기 구문의 `value` 와 일치합니다 . 따라서 `valuePattern` 이후에 나오는 URL 은 다시 작성됩니다 . Portal Server URL 과 게이트웨이 URL 이 앞에 덧붙습니다 . `valuePatterns` 에 대한 자세한 내용은 116 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오 .

```
<input type=hidden name=name3 value="../../html/test.html" >
```

// `name` 이 규칙에서 지정된 `field` 이름에 대응되지 않기 때문에 이 URL 은 다시 작성되지 않습니다 .

```
</FORM>
```

**<FORM name=form2 method=POST action="gateway-URL/http://abc.sesta.com/casestudy/html/form.html">
**

// <Attribute name="action"/> 이 기본 규칙 집합에서 HTML 규칙의 일부로 정의되었기 때문에 이 URL 은 다시 작성됩니다 . URL 이 이미 절대 경로이므로 접두어로 게이트웨이 URL 만 사용하면 됩니다 .

<input type=hidden name=name1 value="0|1234|../../html/test.html" >

// 폼 이름이 규칙에서 지정한 이름과 일치하지 않기 때문에 이 URL 은 다시 작성되지 않습니다 .

</FORM>

</BODY>

RW_END

</HTML>

HTML 애플릿 예제

➤ 애플릿 예제를 사용하려면

1. 애플릿 클래스 파일을 얻습니다 . RewriteURLinApplet.class 파일은 다음 위치에 있습니다 .

   portal-server-URL/rewriter/HTML/applet/appletcode

   애플릿 코드가 있는 페이지의 기본 URL 은 다음과 같습니다 .

   portal-server-URL/rewriter/HTML/applet/rule1.html

2. 이 예제에 지정된 규칙을 "HTML 속성 재작성을 위한 규칙 " 부분의 default_gateway_ruleset 에 추가하십시오 .

3. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 default_gateway_ruleset 을 편집합니다 .

4. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML

<html>

Rewriting starts


```
<applet codebase=appletcode code=RewriteURLinApplet.class archive=/test>
<param name=Test1 value="/index.html">
<param name=Test2 value="../index.html">
<param name=Test3 value="../../index.html">
</applet>
Rewriting ends
</html>
```

규칙

```
<Applet source="*/rule1.html" code="RewriteURLinApplet.class" param="Test*"
/>
```

재작성 후의 HTML

```
<HTML>
Rewriting starts
<BR>
```

<APPLET
codebase=gateway-URL/portal-server-URL/rewriter/HTML/applet/appl
etcode=RewriteURLinApplet.class archive=/test >

// 규칙 `<Attribute name="codebase"/>` 가 이미 `default_gateway_ruleset` 파일의 일부이기 때문에 이 URL 은 다시 작성됩니다 . `appletcode` 디렉토리까지의 경로와 함께 게이트웨이와 Portal Server URL 이 접두어로 사용됩니다 .

<param name=Test1
value="gateway-URL/portal-server-URL/index.html">

// 페이지의 기본 URL 이 `rule1.html` 이고 매개 변수 이름이 규칙에 지정된 매개 변수 `Test*` 에 대응되기 때문에 URL 은 다시 작성됩니다 . `index.html` 이 루트 수준에 있도록 지정되었기 때문에 게이트웨이 및 Portal Server URL 이 직접 접두어로 사용됩니다 .

<param name=Test2
value="gateway-URL/portal-server-URL/rewriter/HTML/index.html">

// 페이지의 기본 URL 이 `rule1.html` 이고 매개 변수 이름이 규칙에 지정된 매개 변수 `Test*` 에 대응되기 때문에 URL 은 다시 작성됩니다 . 필요에 따라 경로가 앞에 덧붙습니다 .

```
<param name=Test3
value="gateway-URL/portal-server-URL/rewriter/index.html">
```

// 페이지의 기본 URL 이 `rule1.html` 이고 매개 변수 이름이 규칙에 지정된 매개 변수 `Test*` 에 대응되기 때문에 URL 은 다시 작성됩니다 . 필요에 따라 경로가 앞에 덧붙습니다 .

```
</APPLET>

Rewriting ends

</HTML>
```

JavaScript 컨텐트 예제

JavaScript URL 변수 예제

➤ JavaScript URL 변수 예제를 사용하려면

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/variables/url/js_urls.html

2. 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 `abc.sesta.com` 이 정의되어 있어야 합니다 .

   정의되어 있지 않으면 직접 연결이 가정되고 게이트웨이 URL 이 앞에 덧붙지 않습니다 .

3. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 `default_gateway_ruleset` 에 추가하십시오 .

4. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 `default_gateway_ruleset` 을 편집합니다 .

5. 규칙을 추가한 경우 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

재작성 전의 HTML 페이지

```
<html>

Rewriting starts

<head>

<title>JavaScript Variable test page</title>

</head>
```

```
<body>
<script LANGUAGE="Javascript">
<!--
//URL Variables
var imgsrc="/tmp/tmp.jpg";
var imgsrc="./tmp/tmp.jpg";
var imgsrc="../tmp/tmp.jpg";
var imgsrc="../../tmp/tmp.jpg";
var imgsrc="http://abc.sesta.com/tmp/tmp.jpg";
var imgsrc="../../../tmp/tmp.jpg";
var imgsrc="tmp/tmp.jpg";
//-->
</SCRIPT>
<br>
Testing JavaScript variables!
<br>
<img src="images/logo.gif">
<br>
Image
</body>
<br>
Rewriting ends
</html>
```

규칙

```
<Variable name=”imgsrc” type="URL"/>
```

재작성 후의 HTML 페이지

```
<html>
Rewriting starts
<head>
<title>JavaScript Variable test page</title>
```

```
</head>

<body>

<script LANGUAGE="Javascript">

<!--

//URL Variables

var imgsrc="gateway-URL/portal-server-URL/tmp/tmp.jpg";

var
imgsrc="gateway-URL/portal-server-URL/rewriter/JavaScript/variab
les/url/tmp/tmp.jpg";

var
imgsrc="gateway-URL/portal-server-URL/rewriter/JavaScript/variab
les/tmp/tmp.jpg";

var
imgsrc="gateway-URL/portal-server-URL/rewriter/JavaScript/tmp/tm
p.jpg";

var imgsrc="gateway-URL/http://abc.sesta.com/tmp/tmp.jpg";

var imgsrc="gateway-URL/portal-server-URL/rewriter/tmp/tmp.jpg";

var
imgsrc="gateway-URL/portal-server-URL/rewriter/JavaScript/variab
les/url/tmp/tmp.jpg";
```

// 위의 모든 URL 은 규칙에 지정된 대로 URL 유형이며 이름이 `imgsrc` 인 JavaScript 변수입니다 . 따라서 게이트웨이 및 Portal Server URL 이 앞에 덧붙습니다 . Portal Server URL 에 이어지는 경로는 필요에 따라 덧붙입니다 .

```
//-->

</SCRIPT>

<br>

Testing JavaScript variables!

<br>

<img src="gateway
URL/portal-server-URL/rewriter/JavaScript/variables/url/images/logo.gif">
```

// `<Attribute name="src"/>` 규칙이 `default_gateway_ruleset` 에 정의되었기 때문에 이 행은 다시 작성됩니다 .

```
<br>
```

```
Image
</body>
<br>
Rewriting ends
</html>
```

JavaScript EXPRESSION 변수 예제

➤ **JavaScript Expression 변수 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/variables/expr/expr.html

2. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 `default_gateway_ruleset` 에 추가하십시오 (아직 없는 경우).

3. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 `default_gateway_ruleset` 을 편집합니다 .

4. 규칙을 추가한 경우 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>
<head>
<title>JavaScript EXPRESSION Variables Test Page</title>
</head>
<body>
<script LANGUAGE="Javascript">
<!--
//Expression variables
var expvar1="images";
var expvar2="/logo.gif";
var expvar = expvar1 + expvar2;
document.write("<A HREF="+expvar+">EXPRESSION</A><P>")
var expvar="/images/logo"+".gif";
```

```
document.write("<A HREF="+expvar+">EXPRESSION</A><P>")
//-->
</SCRIPT>
Testing JavaScript EXPRESSION variables
</body>
</html>
```

규칙

```
<Variable type="EXPRESSION" name="expvar"/>
```

재작성 후의 HTML 페이지

```
<html>
<head>
<title>JavaScript EXPRESSION Variables Test Page</title>
</head>
<body>
<SCRIPT>
```

// Rewriter 는 래퍼 함수 psSRAPRewriter_convert_expression 을 여기에 추가합니다 .

```
</SCRIPT>
<script LANGUAGE="Javascript">
<!--
//Expression variables
var expvar1="images";
var expvar2="/logo.gif";
```

var expvar =psSRAPRewriter_convert_expression(expvar1 + expvar2);

// Rewriter 는 이 명령문의 오른쪽을 JavaScript EXPRESSION 변수인 것으로 인식합니다 . Rewriter 는 서버 쪽에서 이 표현식의 값을 결정할 수 없습니다 . 따라서 , psSRAPRewriter_convert_expression 함수가 표현식 앞에 덧붙습니다 . 이 표현식은 클라이언트 쪽에서 평가되어 필요에 따라 다시 작성됩니다 .

```
document.write("<A HREF="+expvar+">EXPRESSION</A><P>")
```

// 이전 구문에서 다시 작성된 expvar 값이 이 표현식의 값에 도달하기 위해 사용됩니다 . 결과가 유효한 URL 이기 때문에 (그래픽이 샘플의 이 위치에) 링크가 제대로 작동합니다 .

```
var expvar="gateway URL/portal-server-URL/images/logo"+".gif";
```

// Rewriter 는 expvar 의 오른쪽을 문자열 표현식인 것으로 인식합니다 . 이것은 서버 쪽에서 결정할 수 있기 때문에 직접 다시 작성됩니다 .

```
document.write("<A HREF="+expvar+">EXPRESSION</A><P>")
```

// 이전 구문에서 다시 작성된 expvar 값이 이 표현식의 값에 도달하기 위해 사용됩니다 . 결과가 유효한 URL 이 아니기 때문에 (샘플의 이 위치에 그래픽이 없음) 링크가 제대로 작동하지 않습니다 .

```
//-->
</SCRIPT>
Testing JavaScript EXPRESSION variables
</body>
</html>
```

JavaScript DHTML 변수 예제

➤ **JavaScript DHTML 변수 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/variables/dhtml/dhtml.html

2. 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 abc.sesta.com 이 정의되어 있어야 합니다 . 정의되어 있지 않으면 직접 연결이 가정되고 게이트웨이 URL 이 앞에 덧붙지 않습니다 .

3. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 default_gateway_ruleset 에 추가하십시오 (아직 없는 경우). Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 default_gateway_ruleset 을 편집합니다 .

4. 규칙을 추가한 경우 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>
<head>
```

```
<title>JavaScript DHTML Variable Test Page</title>
</head>
<body>
<script LANGUAGE="Javascript">
<!--
//DHTML Var
var dhtmlVar="<a href=../../images/test.html>"
var dhtmlVar="<a href=/../images/test.html>"
var dhtmlVar="<a href=/images/test.html>"
var dhtmlVar="<a href=images/test.html>"
var dhtmlVar="<a href=http://abc.sesta.com/images/test.html>"
var dhtmlVar="<img src=http://abc.sesta.com/images/test.html>"
//-->
</SCRIPT>
<br><br>
Testing DHTML Variables
<br><br>
<img src="images/logo.gif">IMAGE
</body>
</html>
```

규칙

```
<Variable name="DHTML">dhtmlVar</Variable>
```

재작성 후의 HTML 페이지

```
<html>
<head>
<title>JavaScript DHTML Variable Test Page</title>
</head>
<body>
<script LANGUAGE="Javascript">
<!--
```

```
//DHTML Var
```

var dhtmlVar="<a href=gateway-URL/portal-server-URL/rewriter/JavaScript/images/test.html>"

// JavaScript DHTML 규칙이 dhtmlVar 오른쪽을 동적 HTML 컨텐트로 인식합니다 . 따라서 default_gateway_ruleset 파일의 HTML 규칙이 적용됩니다 . 동적 HTML 에는 href 속성이 들어있습니다 . default_gateway_ruleset 은 <Attribute name="href"/> 의 규칙을 정의합니다 . 따라서 href 속성의 값이 다시 작성됩니다 . 하지만 URL 은 절대 경로가 아닙니다 . 따라서 상대 URL 은 페이지의 기본 URL 과 필요한 하위 디렉토리로 대체됩니다 . 여기에 다시 접두어로 게이트웨이 URL 이 사용되어 최종적으로 다시 작성된 결과가 유도됩니다 .

var dhtmlVar="<a href=gateway-URL/portal-server-URL/../images/test.html>"

// 페이지의 기본 URL 이 추가되고 게이트웨이 URL 이 앞에 덧붙여지지만 결과적인 URL 은 올바로 작동하지 않습니다 . 초기 URL /../images/test.html 이 부정확하기 때문입니다 .

var dhtmlVar="<a href=gateway-URL/portal-server-URL/images/test.html>"

// 여기서 다시 JavaScript DHTML 규칙은 오른쪽을 동적 HTML 컨텐트로 인식하고 이를 HTML 규칙으로 전달합니다 . default_gateway_ruleset 의 HTML 규칙 <Attribute name="href"/> 가 적용되며 명령문이 다음과 같이 다시 작성됩니다 . Portal Server URL 과 게이트웨이 URL 이 접두어로 사용됩니다 .

```
var dhtmlVar="<a href=gateway
URL/portal-server-URL/rewriter/JavaScript/variables/dhtml/images/test.html
>"

var dhtmlVar="<a href=gateway URL/http://abc.sesta.com/images/test.html>"
```

var dhtmlVar="<img src=gateway-URL/http://abc.sesta.com/images/test.html>"

// JavaScript DHTML 규칙은 오른쪽의 동적 HTML 컨텐트를 확인하고 문구를 HTML 규칙으로 전달합니다 . default_gateway_ruleset 의 <Attribute name="src"/> 규칙이 적용됩니다 . URL 이 절대 경로이기 때문에 접두어로 게이트웨이 URL 만 사용하면 됩니다 . 이 URL 이 다시 작성되려면 [도메인 및 부속 도메인의 프록시] 목록에 abc.sesta.com 이 정의되어 있어야 합니다 .

```
//-->

</SCRIPT>
```

```
<br><br>

Testing DHTML Variables

<br><br>

<img
src="gateway-URL/portal-server-URL/rewriter/JavaScript/variables
/dhtml/images/logo.gif">

// <Attribute name="src"/> 규칙이 default_gateway_ruleset 에 정의되었기 때문
에 이 행은 다시 작성됩니다 .

<br><br>

Image

</body>

</html>
```

JavaScript DJS 변수 예제

➤ JavaScript DJS 변수 예제를 사용하려면

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/variables/djs/djs.html

2. 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 `abc.sesta.com` 이 정의되어 있어야 합니다 . 정의되어 있지 않으면 직접 연결이 가정되고 게이트웨이 URL 이 앞에 덧붙지 않습니다 .

3. 이 예제에 지정된 두 개의 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 `default_gateway_ruleset` 에 추가하십시오 (아직 없는 경우). Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 `default_gateway_ruleset` 을 편집합니다 .

4. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>

<head>

<title>Dynamic JavaScript Variable Test Page</title>

</head>

<body>
```

```
<script LANGUAGE="Javascript">
<!--
var dJSVar="var dJSimgsrc='/tmp/tmp/jpg';"
var dJSVar="var dJSimgsrc='../../../tmp/tmp/jpg';"
var dJSVar="var dJSimgsrc='http://abc.sesta.com/tmp/tmp/jpg';"
//-->
</SCRIPT>
<br>
Testing Dynamic JavaScript Variables
<br>
<img src="images/logo.gif">
<br>
Image
</body>
</html>
```

규칙

```
<Variable name="dJSVar" type="DJS"/>
<Variable name="dJSimgsrc‟ type=URL"/>
```

재작성 후의 HTML 페이지

```
<html>
<head>
<title>Dynamic JavaScript Variable Test Page</title>
</head>
<body>
<script LANGUAGE="Javascript">
<!--
var dJSVar="var
dJSimgsrc='gateway-URL/portal-server-URL/tmp/tmp/jpg';"
var dJSVar="var
dJSimgsrc='gateway-URL/portal-server-URL/rewriter/tmp/tmp/jpg';"
```

```
var dJSVar="var
dJSimgsrc='gateway-URL/http://abc.sesta.com/tmp/tmp/jpg';"
```

// 위의 모든 구문은 게이트웨이 및 Portal Server URL 로 다시 작성됩니다 . 필요한 경로가 적합하게 앞에 덧붙여집니다 . 첫 번째 규칙은 dJSVar 의 오른쪽을 동적 JavaScript 변수로 파악합니다 . 그 다음에 dJSimgsrc 의 오른쪽을 URL 유형의 JavaScript 변수로 확인하는 두 번째 규칙으로 전달됩니다 . 규칙에 따라 다시 작성됩니다 .

```
//-->
</SCRIPT>
<br>
Testing Dynamic JavaScript Variables
<br>
<img
src="gateway-URL/portal-server-URL/rewriter/JavaScript/variables
/djs/images/logo.gif">
```

// <Attribute name="src"/> 규칙이 default_gateway_ruleset 에 정의되었기 때문에 이 행은 다시 작성됩니다 .

```
<br>
Image
</body>
</html>
```

JavaScript SYSTEM 변수 예제

➤ **JavaScript System 변수 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/variables/system/system.html

2. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 default_gateway_ruleset 에 추가하십시오 (아직 없는 경우).

3. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 default_gateway_ruleset 을 편집합니다 .

4. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>
<head>
<title>JavaScript SYSTEM Variables Test Page</title>
</head>
<body>
<script LANGUAGE="Javascript">
<!--
//SYSTEM Var
alert(window.location.pathname);
//document.write("<A HREF="+window.location.pathname+">SYSTEM</A><P>")
//-->
</SCRIPT>
Testing JavaScript SYSTEM Variables
<br>
This page displays the path where the current page is located when it is
loaded.
</body>
</html>
```

규칙

```
<Variable name="window.location.pathname" type="SYSTEM"/>
```

재작성 후의 HTML

```
<html>
<head>
<title>JavaScript SYSTEM Variables Test Page</title>
</head>
<body>
<SCRIPT>
convertsystem function definition...
</SCRIPT>
```

```
<script LANGUAGE="Javascript">

<!--

//SYSTEM Var

alert(psSRAPRewriter_convert_system(window.location,
window.location.pathname,"window.location"));
```

// Rewriter 는 window.location.pathname 을 JavaScript SYSTEM 변수로 파악합니다 . 이 변수의 값은 서버 쪽에서 결정할 수 없습니다 . 따라서 Rewriter 는 변수 앞에 psSRAPRewriter_convert_pathname 함수를 덧붙입니다 . 이 래퍼 함수는 클라이언트 쪽에서 변수의 값을 결정하고 필요에 따라 다시 작성합니다 .

```
//-->

</SCRIPT>

Testing JavaScript SYSTEM Variables

<br>

This page displays the path where the current page is located when it is
loaded.

</body>

</html>
```

JavaScript URL 함수 예제

➤ **JavaScript URL 함수 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/functions/url/url.html

2. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 default_gateway_ruleset 에 추가하십시오 (아직 없는 경우). Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 default_gateway_ruleset 을 편집합니다 .

3. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>

<body>

JavaScript URL Function Test Page
```

```
<br >
<script language="JavaScript">
<!--
function test(one,two,three)
{
alert(one + "##" + two + "##" +three);
}
test("/test.html","../test.html","123");
window.open("/index.html","gen",width=500,height=500);
//-->
</SCRIPT>
</body>
</html>
```

규칙

```
<Function type="URL" name="test" paramPatterns="y,y"/>
<Function type="URL" name="window.open" paramPatterns="y"/>
```

재작성 후의 HTML 페이지

```
<html>
<body>
JavaScript URL Function Test Page
<br>
<script language="JavaScript">
<!--
function test(one,two,three)
{
alert(one + "##" + two + "##" +three);
}
test("/test.html","../test.html","123");
window.open("gateway-URL/portal-server-URL/index.html","gen",width=500,hei
ght=500);
```

```
//-->
</SCRIPT>
</body>
</html>
```

JavaScript EXPRESSION 함수 예제

➤ **JavaScript Expressions 함수 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/functions/expr/expr.html

2. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 `default_gateway_ruleset` 에 추가하십시오 (아직 없는 경우).

3. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 `default_gateway_ruleset` 을 편집합니다 .

4. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>
<body>
JavaScript EXPRESSION Function Test Page
<br><br><br>
<script language="JavaScript">
<!--
function jstest2()
{
return ".html";
}
function jstest1(one)
{
return one;
}
```

```
var dir="/images/test"
var test1=jstest1(dir+"/test"+jstest2());
document.write("<a HREF="+test1+">Test</a>");
alert(test1);
//-->
</SCRIPT>
</body>
</html>
```

규칙

```
<Function type="EXPRESSION" name="jstest1" paramPatterns="y"/>
```

재작성 후의 HTML 페이지

```
<html>
<body>
JavaScript EXPRESSION Function Test Page
<br><br><br>
<script>
<!--
// psSRAPRewriter_convert_expression을 포함하여 다양한 함수가 여기에 나옵니다.
//-->
</SCRIPT>
<script language="JavaScript">
<!--
function jstest2()
{
return ".html";
}
function jstest1(one)
{
return one;
}
```

```
var dir="/images/test"

var
test1=jstest1(psSRAPRewriter_convert_expression(dir+"/test"+jste
st2()));
```

// 규칙이 EXPRESSION 유형인 jstest1 함수의 첫 번째 매개 변수를 다시 써야 한다고 규정합니다 . 이 표현식의 값은 /test/images/test.html 입니다 . 이 앞에 게이트웨이 및 Portal Server URL 이 덧붙습니다 .

```
document.write("<a HREF="+test1+">Test</a>");

alert(test1);

//-->

</SCRIPT>

</body>

</html>
```

JavaScript DHTML 함수 예제

➤ **JavaScript DHTML 함수 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/functions/dhtml/dhtml.html

2. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 default_gateway_ruleset 에 추가하십시오 (아직 없는 경우).

3. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 default_gateway_ruleset 을 편집합니다 .

4. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>

<head>

Testing JavaScript DHTML Functions

<br>

<br>

<script>
```

```
<!--
document.write('<a href="/index.html">write</a><BR>')
document.writeln('<a href="index.html">writeln</a><BR>')
document.write("http://abc.sesta.com/index.html<BR>")
document.writeln("http://abc.sesta.com/index.html<BR>")
//-->
</SCRIPT>
</head>
<body BGCOLOR=white >
<br><br>
Testing document.write and document.writeln
</body>
</html>
```

규칙

```
<Function type="DHTML" name=" document.write" paramPatterns="y"/>
<Function type="DHTML" name=" document.writeln" paramPatterns="y"/>
```

재작성 후의 HTML 페이지

```
<html>
<head>
Testing JavaScript DHTML Functions
<br>
<br>
<script>
<!--
document.write('<a
href="gateway-URL/portal-server-URL/index.html">write</a><BR>')
```

// 첫 번째 규칙은 DHTML JavaScript 함수 `document.write` 의 첫 번째 매개 변수를 다시 작성해야 한다고 지정합니다 . Rewriter 는 첫 번째 매개 변수를 단순 HTML 명령문으로 파악합니다 . `default_gateway_ruleset` 의 HTML 규칙 부분에는 구문을 다시 써야 한다고 지시하는 `<Attribute name="href" />` 규칙이 있습니다 .

```
document.writeln('<a
href="gateway-URL/portal-server-URL/rewriter/JavaScript/function
s/dhtml/index.html">writeln</a><BR>')
```

// 두 번째 규칙은 DHTML JavaScript 함수 document.writeln 의 첫 번째 매개 변수를 다시 작성해야 한다고 지정합니다 . Rewriter 는 첫 번째 매개 변수를 단순 HTML 명령문으로 파악합니다 . default_gateway_ruleset 의 HTML 규칙 부분에는 구문을 다시 써야 한다고 지시하는 <Attribute name="href" /> 규칙이 있습니다 .

```
document.write("http://abc.sesta.com/index.html<BR>")

document.writeln("http://abc.sesta.com/index.html<BR>")
```

// DHTML 규칙이 함수 document.write 및 document.writeln 을 식별하지만 위의 구문이 다시 작성되지 않습니다 . 이 경우의 첫 번째 매개 변수가 단순 HTML 이 아니기 때문입니다 . 이것은 어떤 문자열도 될 수 있으며 Rewriter 는 이를 다시 작성하는 방법을 알지 못합니다 .

```
//-->

</SCRIPT>

</head>

<body BGCOLOR=white>

<br><br>

Testing document.write and document.writeln

</body>

</html>
```

JavaScript DJS 함수 예제

➤ **JavaScript DJS 함수 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/JavaScript/functions/djs/djs.html

2. 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 abc.sesta.com 이 정의되어 있어야 합니다 .

   정의되어 있지 않으면 직접 연결이 가정되고 게이트웨이 URL 이 앞에 덧붙지 않습니다 .

3. 이 예제에 지정된 규칙을 "JavaScript 소스 재작성을 위한 규칙 " 부분의 default_gateway_ruleset 에 추가하십시오 (아직 없는 경우). Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 default_gateway_ruleset 을 편집합니다 .
4. 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

재작성 전의 HTML 페이지

```
<html>

Test for JavaScript DJS Functions

<br>

<script>

menu.addItem(new NavBarMenuItem("All Available

Information","JavaScript:top.location='http://abc.sesta.com'"));

//menu.addItem(new NavBarMenuItem("All Available
Information","http://abc.sesta.com"));

</script>

</html>
```

규칙

```
<Function type="DJS" name="NavBarMenuItem" paramPatterns=",y"/>

<Variable type="URL" name="top.location"/>
```

재작성 후의 HTML 페이지

```
<html>

Testing JavaScript DJS Functions

<br>

<script>

menu.addItem(new NavBarMenuItem("All Available
Information","javaScript:top.location='gateway-URL/http://abc.se
sta.com'"));
```

// abc.sesta.com 은 게이트웨이 서비스에서 [도메인 및 부속 도메인의 프록시] 목록에 있는 항목입니다 . 따라서 Rewriter 는 이 URL 을 다시 작성해야 합니다 . 그러나 이 값은 절대 URL 이기 때문에 접두어로 Portal Server URL 을 사용할 필요가 없습니다 . DJS 규칙은 DJS 함수 NavBarMenuItem 의 두 번째 매개 변수를 다시 작성해야

한다고 지정합니다 . 하지만 두 번째 매개 변수 역시 JavaScript 변수입니다 . 이 변수의 값을 다시 쓰기 위해 두 번째 규칙이 필요합니다 . 두 번째 규칙은 JavaScript 변수 `top.location` 의 값을 다시 써야 한다고 지정합니다 . 모든 조건이 충족되면 URL 이 다시 작성됩니다 .

```
//menu.addItem(new NavBarMenuItem("All Available
Information","http://abc.sesta.com"));
```

// DJS 규칙에서 함수 `NavBarMenuItem` 의 두 번째 매개 변수를 다시 써야 한다고 지정하고 있지만 이 구문에서는 일어나지 않습니다 . Rewriter 가 두 번째 매개 변수를 단순 HTML 로 인식하지 않기 때문입니다 .

```
</script>
```

```
</html>
```

XML 속성 예제

➤ **XML 속성 예제를 사용하려면**

1. 이 예제는 다음에서 액세스할 수 있습니다 .

   portal-server-URL/rewriter/XML/attrib.html

2. 이 예제에 지정된 규칙을 "XML 소스 재작성을 위한 규칙 " 부분의 `default_gateway_ruleset` 에 추가하십시오 (아직 없는 경우).

3. Identity Server 관리 콘솔에 있는 Portal Server 구성의 Rewriter 서비스에서 `default_gateway_ruleset` 을 편집합니다 .

4. 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

재작성 전의 XML

```
<html>
```

```
RW_START
```

```
<body>
```

```
<xml>
```

```
<baseroot href="/root.html"/>
```

```
</xml>
```

```
<xml>
```

```
<img href="image.html"/>
</xml>
<xml>
<string href="1234|substring.html"/>
</xml>
<xml>
<check href="1234|string.html"/>
</xml>
</body>
RW_END
</html>
```

규칙

```
<Attribute name="href" tag="check" valuePatterns="1234|"/>
```

재작성 후의 HTML

```
<html>
Rewriting starts
<br>
<br>
<body>
<xml><baseroot href="/root.html"/></xml>
<xml><img href="image.html"/></xml>
<xml><string href="1234|substring.html"/></xml>
```

<xml><check href="1234|gateway-URL/portal-server-URL/rewriter/XML/string.html"/></xml>

// 이 명령문은 규칙에 지정된 조건에 대응되기 때문에 다시 작성됩니다. Attribute name 은 href 이고 , tag 는 check 이며 , valuePatterns 는 1234 입니다 . valuePatterns 이후의 문자열이 다시 작성됩니다 . valuePatterns 에 대한 자세한 내용은 116 페이지의 " 규칙에서 패턴 매칭 사용 " 을 참조하십시오 .

```
</body>
Rewriting ends
```

```
</html>
```

사례 연구

여기에는 예제 메일 클라이언트에 대한 소스 HTML 페이지가 포함되어 있습니다. 이 사례 연구에서는 가능한 모든 경우와 규칙을 다루지 않습니다. 실제 인트라넷 페이지의 규칙을 쉽게 구성하도록 돕기 위한 예제 규칙 집합입니다.

가정

이 사례 연구에서는 다음을 가정합니다.

- 메일 클라이언트의 기본 URL 이 `abc.siroe.com` 이라고 가정합니다.
- 게이트웨이 URL 이 `gateway.sesta.com` 이라고 가정합니다.
- 게이트웨이 서비스의 [도메인 및 부속 도메인의 프록시] 목록에 관련 항목이 있습니다.

예제 페이지 1

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

<!-- saved from
url=(0053)http://abc.siroe.com/mailclient/destin/?Cmd=navbar -->

<HTML XMLNS:WM><HEAD>

<META http-equiv=Content-Type content="text/html; CHARSET=utf-8">

<META http-equiv=Pragma content=no-cache>

<META http-equiv=Expires content=0><!--Copyright (c) 2000 Microsoft
Corporation.   All rights reserved.--><!--CURRENT FILE== "IE5" "WIN32"
navbar -->

<STYLE>WM\:DROPMENU {

BEHAVIOR: url(http://abc.siroe.com/mailweb/controls/dropmenu.htc)

}

</STYLE>

<LINK href="destin_files/navbar.css" type=text/css rel=stylesheet>

<SCRIPT language=javascript>

var g_szUserBase= "http://abc.siroe.com/mailclient/destin"+"/";
```

```
var g_szFolder= ".";
var g_szVirtualRoot= "http://abc.siroe.com/mailweb";
var g_szImagePath= g_szVirtualRoot + "/img/";
</SCRIPT>
<SCRIPT src="/destin_files/navbar.js"></SCRIPT>
<META content="MSHTML 6.00.2600.0" name=GENERATOR></HEAD>
<BODY oncontextmenu=return(event.ctrlKey); onselectstart=return(false);
id=outbar_mainbody style="BACKGROUND-COLOR: appworkspace" leftMargin=0
topMargin=0 scroll=no>
<TABLE class=nbTableMain id=nbTableMain style="HEIGHT: 100%" cellSpacing=0
cols=1 cellPadding=0 rows="2">
<TBODY>
<TR>
<TD class=treeBrand>
<DIV class=treeOFLOW><IMG
style="PADDING-RIGHT: 0px; PADDING-LEFT: 0px; PADDING-BOTTOM: 0px;
PADDING-TOP: 0px"
src="/destin_files/logo-ie5.gif" border=0></DIV></TD></TR>
<TR height="100%">
<TD>
<TABLE class=nbTable cellSpacing=0 cols=1 cellPadding=0 rows="4">
<TBODY>
<TR>
<TD class=nbFlybar id=show_navbar onkeydown=flybar_keydown()
onclick=ToggleTab(this.id) tabIndex=0 noWrap>
<DIV class=treeOFLOW>Shortcuts</DIV></TD></TR>
<TR style="HEIGHT: 100%">
<TD id=idOutbarpane style="TEXT-ALIGN: center" vAlign=top><A
id=inbox
```

```
href="http://abc.siroe.com/mailclient/destin/Inbox/?Cmd=contents
&amp;Page=1"
target=viewer alt="Go to inbox"><IMG class=nbImage alt="Go to inbox"
src="destin_files/navbar-inbox.gif"></A>
<DIV class=nbLabel>Inbox</DIV><BR><A id=calendar
href="http://abc.siroe.com/mailclient/destin/Calendar/?Cmd=contents"
target=viewer alt="Go to calendar"><IMG class=nbImage
alt="Go to calendar" src="destin_files/navbar-calendar.gif"></A>
<DIV class=nbLabel>Calendar</DIV><BR><A id=contacts

href="http://abc.siroe.com/mailclient/destin/Contacts/?Cmd=contents"
target=viewer alt="Go to contacts"><IMG class=nbImage
alt="Go to contacts" src="destin_files/navbar-contacts.gif"></A>
<DIV class=nbLabel>Contacts</DIV><BR><A id=options
href="http://abc.siroe.com/mailclient/destin/?Cmd=options"
target=viewer alt="Go to options"><IMG class=nbImage
alt="Go to options" src="destin_files/navbar-options.gif"></A>
<DIV class=nbLabel>Options</DIV></TD></TR>
<TR style="HEIGHT: 1.5em">
<TD class=nbFlybar id=show_folders onkeydown=flybar_keydown()
onclick=ToggleTab(this.id) tabIndex=0 noWrap>
<DIV class=treeOFLOW>Folders</DIV></TD></TR>
<TR>
<TD class=nbTreeProgress id=treeProgress style="DISPLAY: none"
vAlign=top noWrap><SPAN id=idLoading
style="OVERFLOW: hidden">Loading...</SPAN>
</TD></TR></TBODY></TABLE></TD></TR></TBODY></TABLE>
</BODY></HTML>
```

설명

표 3-3 은 예제 규칙 집합과 사례 연구 사이의 매핑을 보여줍니다 .

표 3-3 예제 규칙 집합과 사례 연구 사이의 매핑

페이지 컨텐트	적용 규칙	Rewriter 결과	설명
`var g_szVirtualRoot="http://abc.siroe.com/mailweb";`	<Variable name="URL"> g_szVirtualRoot </Variable>	`var g_szVirtualRoot= "http://gateway.sesta.com/http://abc.siroe.com/mailweb";`	g_szVirtualRoot 는 그 값이 단순 URL 인 변수입니다 . 이 규칙은 URL 유형의 변수 g_szVirtualRoot 를 검색하라고 Rewriter 에 지시합니다 . 웹 페이지에 이런 변수가 있으면 Rewriter 가 이를 절대 URL 로 변환하고 접두어로 게이트웨이 URL 을 사용합니다 .
`src="/destin_files/logo-ie5.gif"`	<Attribute name="src" / >	`src="http://gateway.sesta.com/http://abc.siroe.com/destin_files/logo-ie5.gif`	src 는 속성 이름이며 태그나 valuePattern 이 따라붙지 않습니다 . 이 규칙은 이름이 src 인 모든 속성을 검색하고 그 속성 값을 다시 작성하도록 Rewriter 에 지시합니다 .
`href="http://abc.siroe.com/mailclient/destin/Inbox/?Cmd=contents&amp;Page=1"`	<Attribute name="href" />	`href="http://gateway.sesta.com/http://abc.siroe.com/mailclient/destin/Inbox/?Cmd=contents&amp;Page=1"`	href 는 속성 이름이며 태그나 valuePattern 이 따라붙지 않습니다 . 이 규칙은 이름이 href 인 모든 속성을 검색하고 그 속성 값을 다시 작성하도록 Rewriter 에 지시합니다 .

참고 규칙 집합을 적용하는 우선 순위는 hostname-subdomain-domain 입니다 .

예를 들어 , 도메인 기반 규칙 집합 목록에 다음 항목이 있다고 가정합니다 .

sesta.com|ruleset1

eng.sesta.com|ruleset2

host1.eng.sesta.com|ruleset3

ruleset3 은 host1 의 모든 페이지에 적용됩니다 .

ruleset2 는 host1 에서 가져온 페이지를 제외하고 eng 부속 도메인의 모든 페이지에 적용됩니다 .

ruleset1 은 eng 부속 도메인과 host1 에서 가져온 페이지를 제외하고 sesta.com 도메인의 모든 페이지에 적용됩니다 .

5. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
6. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

Outlook Web Access 의 규칙 집합

SRA 소프트웨어는 Sun Java System Web Server(이전의 Sun™ ONE Web Server) 및 IBM 응용 프로그램 서버에 설치된 MS Exchange 2000 SP3 와 Outlook Web Access (OWA) 의 MS Exchange 2003 을 지원합니다 .

➤ OWA 규칙 집합을 구성하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .

5. [규칙 집합에 URI 매핑] 필드에서 Exchange 2000 이 설치된 서버 이름 , 그리고 이어 Exchange 2000 서비스 팩 4 OWA 규칙 집합의 이름을 입력합니다 .

   예 :

   exchange.domain.com|exchange_2000sp3_owa_ruleset.

6.x 규칙 집합을 3.0 과 매핑

다음 표에서는 SRA Rewriter 규칙과 이전 릴리스의 Portal Server 제품의 매핑을 나열합니다 .

표 3-4 SP3 의 규칙 매핑

Rewriter 6.0 DTD 요소	**Rewriter 3.0 목록 상자 이름**
HTML 컨텐트에 대한 규칙	
속성 - URL	Rewrite HTML 속성
속성 - DJS	JavaScript 를 포함한 Rewrite HTML 속성
폼	Rewrite 폼 입력 태그 목록
애플릿	Rewrite 애플릿 / 개체 매개 변수 값 목록
JavaScript 컨텐트에 대한 규칙	
변수 - URL	URL 의 Rewrite JavaScript 변수
변수 - EXPRESSION	Rewrite JavaScript 변수 함수
변수 - DHTML	HTML 의 Rewrite JavaScript 변수
변수 - DJS	JavaScript 의 Rewrite JavaScript 변수
변수 - SYSTEM	Rewrite JavaScript 시스템 변수
함수 - URL	Rewrite JavaScript 함수 매개 변수
함수 - EXPRESSION	Rewrite JavaScript 함수 매개 변수 함수
함수 - DHTML	HTML 의 Rewrite JavaScript 함수 매개 변수
함수 - DJS	JavaScript 의 Rewrite JavaScript 함수 매개 변수
XML 컨텐트에 대한 규칙	
속성 - URL	XML 문서의 Rewrite 속성
TagText	XML 문서의 Rewrite 텍스트 데이터

표 3-4 SP3 의 규칙 매핑

Rewriter 6.0 DTD 요소	Rewriter 3.0 목록 상자 이름
CSS 컨텐트에 대한 규칙	
규칙이 필요 없습니다 . 기본적으로 모든 URL 이 변환됩니다 .	
WML 컨텐트에 대한 규칙	
정의된 규칙이 없습니다 . WML 은 HTML 로 취급되며 HTML 규칙이 적용됩니다 .	
WMLScript 컨텐트에 대한 규칙	
WML 스크립트에 대한 지원 없음	

4 장

NetFile

이 장에서는 NetFile 과 그 작업에 대해 설명합니다 . NetFile 을 구성하려면 285 페이지 10 장 "NetFile 구성 " 을 참조하십시오 .

이번 장에서는 다음 주제를 다룹니다 .

- NetFile 개요
- 지원되는 파일 액세스 프로토콜
- NetFile 에 대한 액세스 활성화
- NetFile 의 로깅 활성화
- UNIX 인증 구성

NetFile 개요

NetFile 은 사용자가 원격 파일 시스템과 디렉토리에 원격으로 액세스하여 작업할 수 있도록 해주는 파일 관리자 응용 프로그램입니다 .

SRA 의 NetFile 구성 요소는 Java1 및 Java2 애플릿으로 사용할 수 있습니다 . 브라우저에 Java2 플러그인이 없는 사용자는 Java1 애플릿을 사용할 수 있습니다 . Java2 애플릿에는 뛰어난 인터페이스가 있으며 액세스가 더 편합니다 .

NetFile 에는 다음과 같은 주요 기능이 있습니다 .

- 공유 또는 폴더를 추가하거나 제거하는 기능
- 파일 업로드 및 다운로드
- 파일 및 폴더 검색
- GZIP 및 ZIP 을 통한 파일 압축

- NetFile 환경의 우편 기능
- 현재 NetFile 세션 정보 저장
- 파일 끌어서 놓기

NetFile 을 구성하려면 10 장 "NetFile 구성 " 을 참조하십시오 .

지원되는 파일 액세스 프로토콜

NetFile 을 사용하면 FTP, JCIFS(Windows) 및 NFS 프로토콜을 사용하여 원격 시스템에 액세스할 수 있습니다 . NetFile 에는 다음과 같은 파일 액세스 프로토콜 기능도 있습니다 .

- 사용자가 AUTODETECT 를 지정하여 시스템을 추가하면 NetFile 은 다음 시퀀스를 통해 사용할 프로토콜을 자동으로 검색합니다 .
  - 포트 21 에서 FTP 서버용 호스트를 확인합니다 . FTP 응답에 문자열 "NetWare" 가 들어 있으면 NETWARE 호스트로 간주됩니다 .
  - 포트 2049 에서 NFS 서버용 호스트를 확인합니다 .
  - 포트 139 에서 Windows 용 호스트를 확인합니다 .
  - 위의 모든 조치가 실패할 경우 호스트 유형을 결정할 수 없다는 메시지가 표시됩니다 .

  감지되는 첫 번째 파일 시스템 유형이 요청된 호스트에 연결하는 데 사용됩니다. 호스트 감지 순서는 Identity Server 관리 콘솔에서 변경할 수 있습니다 .

참고 서버가 비 표준 포트에서 실행 중이면 연결이 실패합니다 .

- NetFile 을 사용하면 사용자가 원하는 파일 서버와 프로토콜을 선택할 수 있습니다 .

  각 프로토콜에 지원되는 플랫폼이 아래에 나열되어 있습니다 .

표 4-1 파일 시스템 및 지원되는 프로토콜

파일 시스템 / 프로토콜	플랫폼
NFS	Solaris 2.6 이상
JCIFS	Windows 95/98/NT/2000/ME/XP

표 4-1 파일 시스템 및 지원되는 프로토콜

파일 시스템 / 프로토콜	플랫폼
FTP	Novell Netware 의 Novell FTP 5.1 Server Win NT 4.0 의 MS FTP Server 4.0 Win NT 2000 의 MS FTP Server 5.0 Solaris FTP Server WU_FTP 2.6.1 ProFTPD 1.2.8 vsFTPd 1.2.0

참고 Novell Netware 에 대한 지원은 FTP 서버를 통해서만 이루어지며 원시 액세스를 통해서는 이루어지지 않습니다 .

참고 NetFile 을 사용하여 파일을 ProFTPD 서버에 업로드하려면 ProFTPD 서버를 실행하는 호스트의 `proftpd.conf` 파일에서 "AllowStoreRestart" 를 "on" 으로 설정해야 합니다 .

NetFile 에 대한 액세스 활성화

SRA 를 설치할 때 NetFile 서비스는 설치 시 지정한 조직용으로만 등록됩니다 .

➤ **조직 및 사용자에 대해 NetFile 을 활성화하려면**

1. NetFile 서비스를 NetFile 액세스를 필요로 하는 조직에 등록합니다 .
2. NetFile 서비스를 기반으로 NetFile 정책을 만들고 NetFile 에 액세스가 필요한 조직과 역할에 NetFile 정책을 지정합니다 .
3. NetFile 에 액세스가 필요한 각 사용자에게 NetFile 서비스를 지정합니다 .

   정책 및 서비스를 만들기와 지정에 대한 자세한 내용은 *Identity Server 관리 설명서*를 참조하십시오 .

NetFile 의 로깅 활성화

Identity Sever 로깅 서비스를 통해 로그 위치를 지정하면 NetFile 로깅을 활성화할 수 있습니다 . 로그 파일의 이름은 `srapNetFile` 입니다 . 기본적으로 로그 파일은 `/var/opt/SUNWam/logs` 디렉토리에 있습니다 .

UNIX 인증 구성

➤ **Unix 인증을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. 관리 콘솔에서 [Identity 관리] 탭을 선택합니다 .
3. 왼쪽 보기 표시 영역의 [보기] 드롭다운 메뉴에서 [서비스] 를 선택합니다 .

   오른쪽 보기 표시 영역에 UNIX 가 나타나면 이를 등록해야 합니다 .
4. UNIX 옆의 확인란을 선택하고 [등록] 을 눌러 서비스를 등록합니다 .
5. 왼쪽 보기 표시 영역의 UNIX 옆에 있는 화살표를 누르고 [만들기] 를 클릭합니다 .

   서비스 템플리트가 만들어집니다 .
6. [저장] 을 누릅니다 .
7. 관리 콘솔에서 로그아웃합니다 .
8. Identity 서버를 루트 또는 실행하도록 구성된 사용자 Identity 서버로 다시 시작합니다 .

   ```
   /etc/init.d/amserver startall
   ```
9. doUnix 프로세스가 실행되고 있는지 확인합니다 .

   ```
   ps -ef | grep doUnix
   ```

➤ **Unix 인증을 구성하려면**

1. 다음과 같은 구성 포트의 로컬 호스트에 텔넷 연결합니다 .

   ```
   telnet localhost 58946
   ```
2. Unix Helper 수신 포트 번호를 입력합니다 .

   수신 포트에 기본값 `57946` 을 지정합니다 .
3. Unix Helper 세션 시간 초과 값 (초) 을 입력합니다 .

4. Unix Helper 최대 세션 값을 입력합니다 .

   "doUnix 가 성공적으로 구성되었습니다 " 라는 메시지가 표시됩니다 .

5 장

Netlet

이 장에서는 Netlet 을 사용하여 사용자의 원격 데스크탑과 인트라넷에서 응용 프로그램을 실행하는 서버 사이에서 응용 프로그램을 안전하게 실행하는 방법을 설명합니다 . Netlet 을 구성하려면 307 페이지 11 장 "Netlet 구성 " 을 참조하십시오 .

이번 장에서는 다음 주제를 다룹니다 .

- Netlet 개요
- 원격 호스트에서 애플릿 다운로드
- Netlet 규칙 정의
- 샘플 Netlet 규칙
- Netlet 로깅 사용 설정
- Sun Ray 환경에서 Netlet 실행

Netlet 개요

Sun Java™ System Portal Server 소프트웨어 사용자는 원격 데스크탑에서 가장 많이 사용하는 응용 프로그램이나 회사별 응용 프로그램을 안전한 방식으로 실행할 수 있습니다 . 플랫폼에 Netlet 을 설치하면 이 응용 프로그램에 대한 액세스를 보호할 수 있습니다 .

Netlet 을 사용하면 사용자는 인터넷과 같이 안전하지 않은 네트워크에서 공통 TCP/IP 서비스를 안전하게 실행할 수 있습니다 . TCP/IP 응용 프로그램 (텔넷 및 SMTP), HTTP 응용 프로그램 및 고정 포트 응용 프로그램을 실행할 수 있습니다 .

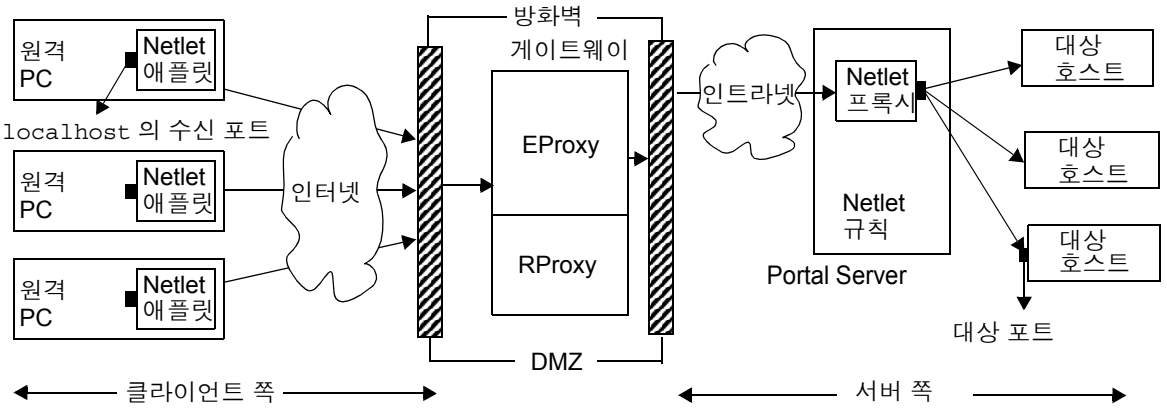
TCP/IP 기반이거나 고정 포트를 사용하는 경우 Netlet에서 응용 프로그램을 실행할 수 있습니다.

참고 동적 포트는 FTP 를 사용할 때만 지원됩니다. Microsoft Exchange 를 사용하려면 Outlook Web Access(OWA) 를 사용합니다.

Netlet 구성 요소

Netlet 에서 사용되는 다양한 구성 요소가 그림 5-1 에 나와 있습니다.

그림 5-1 Netlet 구성 요소



localhost 의 수신 포트

Netlet 애플릿이 수신하는 클라이언트 컴퓨터에 있는 포트입니다. 클라이언트 컴퓨터는 localhost 입니다.

Netlet 애플릿

Netlet 애플릿은 원격 클라이언트 컴퓨터와 텔넷, Grphon 또는 Citrix 와 같은 인트라넷 응용프로그램 사이에 암호화된 TCP/IP 터널을 설정하는 역할을 합니다. 애플릿은 패킷을 암호화하여 이를 게이트웨이로 보낸 다음, 게이트웨이로부터 받은 응답 패킷의 암호를 해독하여 로컬 응용프로그램으로 보냅니다.

정적 규칙의 경우 Netlet 애플릿은 사용자가 포털에 로그인하면 자동으로 다운로드됩니다. 동적 규칙의 경우, 사용자가 동적 규칙에 해당하는 링크를 누르면 애플릿이 다운로드됩니다. 정적 규칙과 동적 규칙에 대한 자세한 내용은 191 페이지의 " 규칙의 유형 " 을 참조하십시오.

Sun Ray 환경에서 Netlet 을 실행하려면 203 페이지의 "Sun Ray 환경에서 Netlet 실행 " 을 참조하십시오.

Netlet 규칙

Netlet 규칙은 클라이언트 시스템에서 실행해야 하는 응용 프로그램을 해당 대상 호스트로 매핑합니다. 이는 Netlet 규칙에 정의된 포트로 패킷이 전송되는 경우에만 Netlet 이 작동한다는 것을 의미합니다. 이 옵션을 설정하면 보안이 강화됩니다.

관리자로서 Netlet 이 제 기능을 발휘하도록 하려면 특정 규칙을 구성해야 합니다. 이 규칙은 사용할 암호, 불러올 URL, 다운로드할 애플릿, 대상 포트 및 대상 호스트와 같은 다양한 상세 정보를 지정합니다. 클라이언트 시스템에 있는 사용자가 Netlet 을 통해 요청하면 이 규칙에 의해 연결 설정 방식이 결정됩니다. 자세한 내용은 187 페이지의 "Netlet 규칙 정의 " 를 참조하십시오.

Netlet 공급자

Netlet 의 UI 구성 요소입니다. 공급자는 사용자가 Portal Server, Secure Remote Access 데스크탑에서 필요한 응용프로그램을 구성할 수 있도록 해줍니다. 공급자에서 링크를 만든 후 사용자가 그 링크를 누르면 필요한 응용프로그램 실행됩니다. 사용자는 Netlet 공급자의 데스크탑에서 동적 규칙의 대상 호스트를 지정할 수도 있습니다. 187 페이지의 "Netlet 규칙 정의 " 을 참조하십시오.

프록시

모든 클라이언트 요청은 Eproxy 를 통해 라우팅됩니다. Eproxy 는 Netlet 요청만 처리하고 다른 요청은 RProxy 로 전달합니다. Eproxy 는 Netlet 요청을 구문 분석하여 Netlet 프록시로 전달하거나 (활성화된 경우) 대상 호스트로 직접 전달합니다.

Netlet 프록시 (선택 사항)

게이트웨이는 원격 클라이언트 컴퓨터와 게이트웨이 사이에 보안 터널을 보장합니다. Netlet 프록시는 선택 사항이며 설치 중에 이 프록시의 설치는 선택하지 않아도 됩니다. Netlet 프록시에 대한 자세한 내용은 65 페이지의 "Netlet 프록시 사용 " 을 참조하십시오.

대상 포트

대상 응용 프로그램의 호스트가 수신하는 포트입니다 .

Netlet 사용 시나리오

Netlet 을 사용하는 경우 다음 이벤트 시퀀스가 사용됩니다 .

1. 원격 사용자가 Portal Server, Secure Remote Access 데스크탑에 로그인합니다 .
2. 사용자 , 역할 또는 조직에 정적 Netlet 규칙이 정의되어 있으면 Netlet 애플릿이 원격 클라이언트에게 자동으로 다운로드됩니다 .

   사용자 , 역할 또는 조직에 동적 규칙이 정의되어 있으면 사용자는 Netlet 공급자에서 필요한 응용프로그램을 구성해야 합니다 . 사용자가 Netlet 공급자에서 응용프로그램 링크를 누르면 Netlet 애플릿이 다운로드됩니다 . 정적 규칙과 동적 규칙에 대한 자세한 내용은 187 페이지의 "Netlet 규칙 정의 " 를 참조하십시오 .
3. Netlet 이 Netlet 규칙에 정의된 로컬 포트에서 수신합니다 .
4. Netlet 이 Netlet 규칙에 지정된 포트에서 원격 클라이언트와 호스트 사이의 채널을 설정합니다 .

Netlet 작업

Netlet 이 여러 조직에 있는 다양한 사용자의 필요에 따라 작동할 수 있으려면 다음을 수행해야 합니다 .

1. 사용자 요구사항에 따라 정적 규칙이나 동적 규칙 중 어느 것을 만들어야 할지 결정합니다 . 191 페이지의 " 규칙의 유형 " 을 참조하십시오 .
2. Identity Server 관리 콘솔의 [서비스 구성] 탭의 Netlet 템플리트에서 전역 옵션을 정의합니다 . 307 페이지 11 장 "Netlet 구성 " 을 참조하십시오 .
3. 규칙이 조직 , 역할 또는 사용자 중 어디에 기반할지 결정하고 각 수준에서 필요에 따라 수정합니다 . 조직 , 역할 및 사용자에 대한 상세 정보는 *Portal Server 관리 설명서*를 참조하십시오 .

참고 `srapNetletServlet.properties` 파일에서 프레임 집합 매개 변수의 값을 현지화하지 마십시오 .

원격 호스트에서 애플릿 다운로드

원격 시스템에서 가져와야 하는 내장된 애플릿이 포함된 URL 에서 페이지가 반환되는 경우가 있습니다 . 하지만 Java 보안에서는 애플릿을 다운로드한 호스트가 아닌 호스트와 애플릿이 통신하는 것을 허용하지 않습니다 . 애플릿이 로컬 네트워크 포트를 통해 게이트웨이와 통신할 수 있도록 허용하려면 Identity Server 관리 콘솔에서 [애플릿 다운로드] 필드를 선택하고 다음 구문을 지정해야 합니다 .

local-port:server-host:server-port

여기서

local-port 는 Netlet 이 애플릿으로부터 오는 트래픽을 수신하는 로컬 포트입니다 .

server-host 는 애플릿을 다운로드하는 위치입니다 .

server-port 는 애플릿 다운로드에 사용되는 포트입니다 .

Netlet 규칙 정의

Netlet 구성은 SRA 구성 섹션에서 Identity Server 관리 콘솔에 구성된 Netlet 규칙으로 정의됩니다 . Netlet 규칙은 조직 , 역할 또는 사용자용으로 구성할 수 있습니다 . Netlet 규칙이 역할이나 사용자용인 경우 조직을 선택한 다음 원하는 역할이나 사용자를 선택합니다 .

주의 Netlet 규칙은 멀티바이트 항목을 지원하지 않습니다 . Netlet 규칙의 편집 가능한 모든 필드에 멀티바이트 문자를 지정하지 마십시오 .

Netlet 규칙에는 64000 보다 큰 포트 번호가 포함되면 안됩니다 .

표 5-1 에는 Netlet 규칙의 필드가 나열되어 있습니다 .

표 5-1 Netlet 규칙의 필드

매개 변수	설명	값
규칙 이름	이 Netlet 규칙의 이름을 지정합니다 . 각 규칙마다 고유한 이름을 지정해야 합니다 . 특정 규칙에 대한 사용자 액세스를 정의할 때 유용합니다 . 자세한 내용은 316 페이지의 "Netlet 규칙에 대한 액세스 정의 " 를 참조하십시오 .	
암호화 암호	암호화 암호를 정의하거나 사용자 선택 가능한 암호 목록을 지정합니다 .	선택한 암호는 Netlet 공급자에 목록으로 나타납니다. 사용자는 선택된 목록에서 필요한 암호를 선택할 수 있습니다 . 기본값 - Netlet 관리 콘솔에 지정된 기본 VM 원시 암호와 기본 Java 플러그인 암호가 사용됩니다 .
URL	사용자가 Netlet 공급자에서 관련 링크를 클릭할 때 브라우저에서 여는 URL 을 지정합니다 . 브라우저는 응용프로그램 창을 열고 이 규칙의 뒤 부분에 지정된 로컬 포트 번호에서 `localhost` 에 연결합니다 . 관련 URL 을 지정해야 합니다 .	Netlet 규칙에서 불러온 응용프로그램에 대한 URL. 예 : `telnet://localhost:30000`. 응용프로그램이 애플릿을 사용하여 응용프로그램을 불러오는 경우 URL 을 지정합니다 . `null` - 응용프로그램이 URL 에 의해 시작되지 않거나 데스크탑에서 제어되지 않는 경우 사용자가 설정한 값 . 일반적으로 웹 기반이 아닌 응용프로그램에는 true 입니다 .

표 5-1 Netlet 규칙의 필드

매개 변수	설명	값
애플릿 다운로드	이 규칙에 대한 애플릿 다운로드가 필요한지 여부를 나타냅니다 .	False - 애플릿을 다운로드하지 마십시오 . True - 루프백 포트를 사용하여 Portal Server 컴퓨터에서 애플릿을 다운로드합니다 . *local-port:server-host:server-port* 와 같은 형식으로 애플릿 상세 정보를 지정합니다 . 여기서 • *local-port* 는 클라이언트의 대상 포트를 나타냅니다 . 이 포트는 기본 루프백 포트와 달라야 합니다 . 자세한 내용은 312 페이지의 " 기본 루프백 포트 할당 " 을 참조하십시오 . 각 규칙에 고유한 `local port` 를 지정합니다 . • *server-host* 는 애플릿을 다운로드할 서버의 이름입니다 . • *server-port* 는 애플릿 다운로드에 사용되는 서버의 포트를 나타냅니다 . 애플릿을 다운로드해야 할 경우 서버가 지정되어 있지 않으면 애플릿은 Portal Server 호스트로부터 다운로드됩니다 .
세션 확장	Netlet 이 활성 상태일 때 Portal Server 세션의 유휴 시간 초과를 제어합니다 .	사용 - Netlet 만 활성 상태이고 포털 응용 프로그램의 나머지가 유휴 상태일 때 포털 세션의 유지에 필요합니다 . 사용 안 함 ñ Netlet 응용 프로그램이 활성 상태이지만 포털 응용 프로그램의 나머지가 유휴 상태라도 세션 유휴 시간 초과에서 포털 세션 유휴 시간 초과가 이루어집니다 .
로컬 포트	Netlet 이 수신하는 클라이언트의 포트	*local-port*의 값은 고유해야 합니다. 2개 이상 규칙에서 특정 포트 번호를 지정할 수 없습니다 . 다중 연결을 위한 다중 호스트를 지정하는 경우에는 다중 로컬 포트를 지정합니다 . 구문은 195 페이지의 " 다중 호스트 연결이 있는 정적 규칙 " 을 참조하십시오 . FTP 규칙의 경우 로컬 포트 값이 30021 이어야 합니다 .

표 5-1 Netlet 규칙의 필드

매개 변수	설명	값
대상 호스트	Netlet 연결의 수신자 .	*host* - Netlet 연결을 수신하는 호스트의 이름 . 이 값은 정적 규칙에 사용됩니다 . `siroe` 와 같은 간단한 호스트 이름이나 `siroe.mycompany.com` 과 같은 완전한 정규 DNS 스타일 호스트 이름을 사용합니다 . 다중 호스트를 지정하는 이유는 다음과 같습니다 . • 지정된 각 호스트와 연결을 설정하는 경우 . 지정된 각 호스트에 해당하는 클라이언트 및 대상 포트를 지정해야 합니다 . 구문은 195 페이지의 " 다중 호스트 연결이 있는 정적 규칙 " 을 참조하십시오 . • 지정된 호스트 목록에서 임의의 사용 가능한 호스트에 연결을 시도하는 경우 . 구문은 196 페이지의 " 다중 호스트 선택이 가능한 정적 규칙 " 을 참조하십시오 . `TARGET` - 구문에서 `TARGET` 을 지정하는 규칙은 동적 규칙입니다 . `TARGET` 은 최종 사용자가 데스크탑의 Netlet 공급자에서 필요한 대상 호스트 (하나 또는 여러 개) 를 지정할 수 있음을 나타냅니다 . 단일 규칙에 정적 호스트와 TARGET 을 조합할 수는 없습니다 .
대상 포트	대상 호스트의 포트	호스트 및 대상 호스트 뿐 아니라 대상 포트도 지정해야 합니다 . 다중 대상 호스트가 있는 경우에는 다중 대상 포트를 지정할 수 있습니다 . `port1+port2+port3-port4+port5` 와 같은 형식으로 다중 포트를 지정합니다 . 포트 번호 사이의 플러스 (+) 기호는 한 대상 호스트에 대체 포트가 있음을 나타냅니다 . 포트 번호 사이의 마이너스 (-) 기호는 여러 대상 호스트의 포트 번호를 구분하는 기호입니다 . 여기서 Netlet 은 `port1`, `port2` 및 `port3` 을 순서대로 사용하여 지정된 첫 번째 대상 호스트에 연결을 시도합니다 . 연결이 실패하면 Netlet 은 `port4` 와 `port5` 를 순서대로 사용하여 두 번째 호스트에 연결을 시도합니다 . 정적 규칙에만 다중 포트를 구성할 수 있습니다 .

게이트웨이에서 Portal Server 의 세션 알림을 수신하려면 다음을 추가합니다 .

```
com.iplanet.am.jassproxy.trustAllServerCerts=true
```

아래의 속성 파일에 추가합니다 .

Portal Server 의 `/etc/opt/SUNWam/config/AMConfig.properties`.

2. Identity Server 를 다시 시작합니다 .

규칙의 유형

대상 호스트가 어떻게 규칙에 지정되어 있는지에 따라 Netlet 규칙에는 2 가지 유형이 있습니다 .

정적 규칙

정적 규칙은 대상 호스트를 규칙의 일부로 지정합니다 . 정적 규칙을 만드는 경우 사용자는 필요한 대상 호스트를 지정하지 못합니다 . 다음 예제에서 `sesta` 는 대상 호스트입니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
ftpstatic	SSL_RSA_WITH_RC4_128_MD5	null	false	true	30021	sesta	21

다중 대상 호스트와 포트는 정적 규칙에만 구성할 수 있습니다 . 예는 195 페이지의 " 다중 호스트 연결이 있는 정적 규칙 " 을 참조하십시오 .

동적 규칙

동적 규칙에서는 대상 호스트가 규칙의 일부로 지정되지 않으며 사용자가 Netlet 공급자에서 필요한 대상 호스트를 지정할 수 있습니다 . 다음 예제에서 `TARGET` 은 대상 호스트의 자리 표시자를 말합니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
ftpdynamic	SSL_RSA_WITH_RC4_128_MD5	null	false	true	30021	TARGET	21

암호화 암호

암호화 암호를 기준으로 Netlet 규칙은 다음과 같이 세분화될 수 있습니다 .

- **사용자 구성 가능 암호 규칙** - 이 규칙에서는 사용자가 선택할 수 있는 암호 목록을 지정할 수 있습니다 . 암호 옵션이 Netlet 공급자에 목록으로 나타납니다 . 사용자는 목록에서 필요한 암호를 선택할 수 있습니다 . 다음 예제에서 사용자는 여러 암호 중에서 선택할 수 있습니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
텔넷	SSL_RSA_WITH_RC4_128_SHA SSL_RSA_WITH_RC4_128_MD5	null	false	true	30000	TARGET	23

참고 Portal Server 호스트에 사용 가능하도록 설정된 다양한 비밀번호가 있을 수도 있으나 사용자는 Netlet 규칙의 일부로 구성된 목록에서만 선택할 수 있습니다 .

Netlet 에서 지원되는 암호의 목록은 193 페이지의 " 지원되는 비밀번호 " 를 참조하십시오 .

- **관리자 구성 암호 규칙** - 이 규칙에서는 암호가 Netlet 규칙의 일부로 정의됩니다 . 사용자는 필요한 암호를 선택할 수 없습니다 . 다음 예제에서는 암호가 SSL_RSA_WITH_RC4_128_MD5 로 구성되어 있습니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
텔넷	SSL_RSA_WITH_RC4_128_MD5	null	false	true	30000	TARGET	23

Netlet 에서 지원하는 암호의 목록은 193 페이지의 " 지원되는 비밀번호 " 를 참조하십시오 .

지원되는 비밀번호

표 5-2 에는 Netlet 에서 지원되는 암호가 나열되어 있습니다 .

표 5-2 지원되는 암호 목록

암호
원시 VM 암호
KSSL_SSL3_RSA_WITH_3DES_EDE_CBC_SHA
KSSL_SSL3_RSA_WITH_RC4_128_MD5
KSSL_SSL3_RSA_WITH_RC4_128_SHA
KSSL_SSL3_RSA_EXPORT_WITH_RC4_40_MD5
KSSL_SSL3_RSA_WITH_DES_CBC_SHA
Java 플러그인 암호
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_RSA_WITH_RC4_128_MD5
SSL_RSA_WITH_RC4_128_SHA
SSL_RSA_EXPORT_WITH_RC4_40_MD5
SSL_RSA_WITH_DES_CBC_SHA
SSL_RSA_WITH_NULL_MD5

이전 버전과의 호환성

Portal Server 의 이전 버전에서는 Netlet 규칙의 일부로 암호를 지원하지 않습니다 . 암호가 없는 기존 규칙과의 이전 호환성을 위해 규칙에서는 기본 암호를 사용합니다. 다음과 같이 암호가 없는 기존 규칙은

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
텔넷		telnet://localhost:30000	false	true	30000	TARGET	23

아래와 같이 해석됩니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
텔넷	기본 암호	telnet://localhost:30000	false	true	30000	TARGET	23

이는 암호화 암호 필드가 기본값으로 선택된 관리자 구성 규칙과 비슷합니다 . 자세한 내용은 311 페이지의 " 기본 암호 지정 " 을 참조하십시오 .

참고 Netlet 규칙에는 64000 보다 큰 포트 번호가 포함되면 안됩니다 .

Netlet 규칙 예제

이 부분에서는 Netlet 구문의 원리를 설명하기 위해 몇 가지 Netlet 규칙의 예제가 나와 있습니다 .

- 기본 정적 규칙
- 다중 호스트 연결이 있는 정적 규칙
- URL 을 불러오기 위한 동적 규칙
- 애플릿을 다운로드하기 위한 동적 규칙

기본 정적 규칙

이 규칙은 클라이언트에서 컴퓨터 `sesta` 로 가는 텔넷 연결을 지원합니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
myrule	SSL_RSA_WITH_RC4_128_MD5	null	false	true	1111	sesta	23

여기서

`myrule` 은 규칙의 이름입니다 .

SSL_RSA_WITH_RC4_128_MD5 는 사용할 암호를 나타냅니다 .

null 은 이 응용프로그램이 URL 에 의해 호출되었거나 데스크탑을 통해 실행되지 않음을 나타냅니다 .

false 는 클라이언트가 이 응용프로그램을 실행하기 위해 애플릿을 다운로드하지 않음을 나타냅니다 .

true 는 Netlet 연결이 활성 상태인 경우 Portal Server 에서 시간 초과가 발생하면 안 됨을 나타냅니다 .

1111 은 Netlet 이 대상 호스트로부터 오는 연결 요청을 수신하는 클라이언트의 포트입니다 .

sesta 는 텔넷 연결의 수신자 호스트 이름입니다 .

23 은 연결에 사용되는 대상 호스트의 포트 번호이며 이 경우에는 텔넷에 잘 알려진 포트입니다 .

데스크탑 Netlet 공급자는 링크를 표시하지 않지만 Netlet 은 자동으로 시작되어 지정된 포트 (1111) 에서 수신합니다 . 사용자에게 클라이언트 소프트웨어를 시작하라고 지시하십시오 . 이 경우에는 포트 1111 의 localhost 에 연결하는 텔넷 세션을 시작합니다 .

예를 들어 , 텔넷 세션을 시작하려면 클라이언트는 단말기의 UNIX 명령줄에 다음을 입력해야 합니다 .

```
telnet localhost 1111
```

다중 호스트 연결이 있는 정적 규칙

이 규칙은 클라이언트에서 2 대의 컴퓨터 sesta 및 siroe 로 가는 텔넷 연결을 지원합니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
myrule	SSL_RSA_WITH_RC4_128_MD5	null	false	true	1111	sesta	23
					1234	siroe	23

여기서

23 은 연결에 사용할 대상 호스트의 포트 번호로 , 텔넷용으로 예약된 포트입니다 .

1111 은 Netlet 에서 첫 번째 대상 호스트 sesta 로부터 오는 연결 요청을 수신하는 클라이언트의 포트입니다 .

1234 는 Netlet 에서 두 번째 대상 호스트 siroe 로부터 오는 연결 요청을 수신하는 클라이언트의 포트입니다 .

규칙의 처음 6 개 필드는 194 페이지의 " 기본 정적 규칙 " 에서와 동일합니다 . 차이는 두 번째 대상 호스트를 식별하는 3 개의 추가 필드가 있다는 점입니다 .

규칙에 대상을 추가할 때는 각각의 새로운 대상 호스트에 local port, destination host 및 destination port 필드를 추가해야 합니다 .

참고 각 대상 호스트에 대한 연결을 설명하는 3 개의 필드 집합은 여러 개가 있을 수 있습니다 . 원격 클라이언트가 UNIX 기반인 경우 번호가 낮은 포트는 제한되고 수신기를 시작할 수 있으려면 루트여야 하므로 2048 보다 작은 수신 포트 번호는 사용하면 안됩니다 .

이 규칙은 앞의 규칙과 동일하게 적용됩니다 . Netlet 공급자는 어떠한 링크도 표시하지 않지만 Netlet 은 자동으로 시작되어 지정된 2 개의 포트 (1111 과 1234) 에서 수신합니다 . 사용자는 클라이언트 소프트웨어를 시작해야 합니다 . 이 경우 포트 1111 의 localhost 나 포트 1234 의 localhost 에 연결하는 텔넷 세션을 시작하여 호스트 예제 2 에 연결해야 합니다 .

다중 호스트 선택이 가능한 정적 규칙

다중 대체 호스트를 지정하려면 이 규칙을 사용합니다 . 첫 번째 호스트에 대한 규칙의 연결이 실패하면 Netlet 은 지정된 두 번째 호스트에 연결을 시도합니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
gojoe	SSL_RSA_WITH_RC4_128_MD5	/gojoe.html	8000:gojoeserver:8080	true	10491	siroe+sesta	35+26+491-35+491

여기서

10491 은 Netlet 에서 대상 호스트로부터 오는 연결 요청을 수신하는 클라이언트의 포트입니다 .

Netlet 은 사용 가능한 포트에 따라 포트 35, 포트 26 및 포트 491 에서 같은 순서로 siroe 에 연결을 시도합니다 .

siroe 에 연결할 수 없으면 Netlet 은 포트 35 및 491 에서 같은 순서로 sesta 에 연결을 시도합니다 .

호스트 사이의 플러스 (+) 기호는 대체 호스트를 나타냅니다 .

포트 번호 사이의 플러스 (+) 기호는 한 대상 호스트에 대체 포트가 있음을 나타냅니다 .

포트 번호 사이의 마이너스 (-) 기호는 여러 대상 호스트의 포트 번호를 구분하는 기호입니다 .

URL 을 불러오기 위한 동적 규칙

이 규칙을 사용하면 사용자가 필요한 대상 호스트를 구성하여 Netlet 을 통해 다양한 호스트에 텔넷 연결을 수행할 수 있습니다 .

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
myrule	SSL_RSA_WITH_RC4_128_MD5	telnet://localhost:30000	false	true	30000	TARGET	23

여기서

myrule 은 규칙의 이름입니다 .

SSL_RSA_WITH_RC4_128_MD5 는 사용할 암호를 나타냅니다 .

telnet://localhost:30000 은 규칙에서 불러오는 URL 입니다 .

false 는 애플릿이 다운로드되지 않음을 나타냅니다 .

Extend Session(true) 는 Netlet 연결이 활성 상태이면 Portal Server 에서 시간 초과가 발생하면 안 됨을 나타냅니다 .

30000 은 Netlet 에서 이 규칙에 대한 연결 요청을 수신하는 클라이언트의 포트입니다 .

TARGET 은 사용자가 Netlet 공급자를 사용하여 대상 서버를 구성해야 함을 나타냅니다 .

23 은 Netlet 에서 열린 대상 호스트의 포트 번호이며 , 이 경우에는 텔넷에 잘 알려진 포트입니다 .

➤ 규칙을 추가한 후 Netlet 을 실행하려면

이 규칙을 추가한 후 Netlet 이 예상대로 실행되도록 하려면 사용자는 몇 가지 단계를 완료해야 합니다 . 사용자는 클라이언트 쪽에 다음을 수행해야 합니다 .

1. Portal Server 데스크탑의 Netlet 공급자 섹션에서 [편집] 을 누릅니다 .

   새 Netlet 규칙이 [새 대상 추가] 섹션의 [규칙 이름] 에 나열됩니다 .

2. 규칙 이름을 선택하고 대상 호스트의 이름을 입력합니다 .
3. 변경 사항을 저장합니다 .

   새 링크가 Netlet 공급자 섹션에 표시되 있는 상태로 사용자는 데스크탑으로 돌아갑니다 .

4. 새 링크를 클릭합니다 .

   Netlet 규칙에 주어진 URL 로 이동하는 새 브라우저가 시작됩니다 .

참고 이 단계를 반복하여 같은 규칙에 대상 호스트를 2 개 이상 추가할 수 있습니다 .

애플릿을 다운로드하기 위한 동적 규칙

이 규칙은 클라이언트에서 동적 할당된 호스트로의 GO-Joe 연결을 정의합니다 . 규칙은 애플릿이 있는 서버에서 클라이언트로 GO-Joe 애플릿을 다운로드하게 됩니다.

규칙 이름	암호화 암호	URL	애플릿 다운로드	세션 확장	로컬 포트	대상 호스트	대상 포트
gojoe	SSL_RSA_WITH_RC4_128_MD5	/gojoe.html	8000:gojoeserve:8080	true	3399	TARGET	58

여기서

gojoe 는 규칙의 이름입니다 .

SSL_RSA_WITH_RC4_128_MD5 는 사용할 암호를 나타냅니다 .

예를 들어 /gojoe.html 은 애플릿이 있는 HTML 페이지의 경로이고 이 경로는 포털이 배포된 웹 컨테이너의 문서 루트에 상대적인 값이어야 합니다 .

8000:server:8080 은 포트 8000 은 애플릿을 수신하는 클라이언트의 대상 포트임을 나타내며 gojoeserve 는 애플릿을 제공하는 서버의 이름이고 , 8080 은 애플릿을 다운로드할 서버의 포트입니다 .

Extended Session (true) 는 Netlet 연결이 활성 상태이면 Portal Server 에서 시간 초과가 발생하면 안 되는 것을 나타냅니다 .

3399 는 Netlet 에서 이 유형의 연결 요청을 수신하는 클라이언트의 포트입니다 .

TARGET 은 사용자가 Netlet 공급자를 사용하여 대상 호스트를 구성해야 함을 나타냅니다 .

58 은 Netlet 에서 열린 대상 호스트의 포트 번호이며 , 이 경우에는 GoJoe 용 포트입니다 . 포트 58 은 대상 호스트가 자체 트래픽과 관련하여 수신하는 포트입니다 . Netlet 은 새 애플릿에서 이 포트로 정보를 전달합니다 .

샘플 Netlet 규칙

표 5-3 에는 몇 가지 일반 응용프로그램을 위한 예제 Netlet 규칙이 나열되어 있습니다 .

이 테이블에는 Netlet 규칙의 규칙 이름 , URL, 애플릿 다운로드 , 로컬 포트 , 대상 호스트 , 대상 포트 필드에 해당하는 7 개의 열이 있습니다 . 마지막 열에는 규칙에 대한 설명이 나와 있습니다 .

참고 표 5-3 에서는 Netlet 규칙의 암호와 세션 확장 필드는 나열하지 않습니다 . 제시된 예제에 대해 이 필드 값을 "SSL_RSA_WITH_RC4_128_MD5" 및 "true" 라고 가정하십시오 .

표 5-3 예제 Netlet 규칙

규칙	URL	애플릿 다운로드	로컬 포트	대상 호스트	대상 포트	설명
IMAP	null	false	10143	imapserver	143	클라이언트 쪽의 Netlet `local port` 는 서버 쪽의 `destination port` 와 같지 않아도 됩니다 . 표준 IMAP 및 SMTP 포트 이외의 포트를 사용할 경우 클라이언트를 표준 포트가 아닌 포트에 연결하도록 구성해야 합니다 . Solaris 클라이언트 사용자는 루트로 실행하고 있지 않으면 1024 보다 작은 포트 번호에 연결하는 데 어려움이 있습니다 .
SMTP	null	false	10025	smtpserver	25	
Lotus 웹 클라이언트	null	false	80	lotus-server	80	이 규칙은 Netlet 에 포트 80 에서 클라이언트에서 수신하도록 지시하고 포트 80 의 서버인 lotus-server 에 연결합니다 . Lotus 웹 클라이언트의 요구 사항은 클라이언트 수신 포트가 서버 포트와 일치해야 한다는 것입니다 .
Lotus Notes 비 웹 클라이언트	null	false	1352	lotus-domino	1352	이 규칙을 사용하여 Lotus Notes 클라이언트는 Netlet 을 통해 Lotus Domino 서버에 연결할 수 있습니다 . 클라이언트에서 서버로 연결을 시도할 때는 서버 이름으로 `localhost` 를 지정하면 안됩니다 . Lotus Domino 서버의 실제 서버 이름을 지정해야 합니다 . 서버 이름은 서버의 시스템 이름과 같아야 합니다 . 클라이언트는 Netlet 을 사용할 때 이름을 `127.0.0.1` 로 설정해야 합니다 . 이름을 이렇게 지정하려면 다음 2 가지 방법을 사용합니다 . • 클라이언트 호스트 테이블에서 서버 이름이 `127.0.0.1` 이 되도록 설정합니다 . • `127.0.0.1` 을 가리키는 서버 이름의 DNS 항목을 내보냅니다 . 서버 이름은 설치 시 Domino 서버를 구성하는 데 사용한 서버 이름과 같아야 합니다 .

표 5-3 예제 Netlet 규칙

규칙	URL	애플릿 다운로드	로컬 포트	대상 호스트	대상 포트	설명
Microsoft Outlook 및 Exchange 서버 Windows NT, 2000 및 XP 에서 작동합니다 . Windows NT, 2000 및 XP 용 Rewriter 를 통해 Outlook Web Access 를 사용하십시오	null	false	135	exchange	135	이 규칙은 Netlet 에 클라이언트의 포트 135 에서 수신하여 포트 135 에서 서버 exchange 에 연결하라고 지시합니다 . Outlook 클라이언트는 이 포트를 통해 Exchange 서버에 최초로 연결을 시도하고 서버와 통신하는 데 사용할 후속 포트를 결정합니다 . 클라이언트 컴퓨터 : • 사용자는 Outlook 클라이언트에 구성된 Exchange 서버의 호스트 이름을 localhost 로 변경해야 합니다 . 이 옵션의 위치는 Outlook 버전에 따라 다릅니다 . • 사용자는 호스트 파일을 사용하여 Exchange 서버의 호스트 이름 (단일 및 완전한 정규) 을 IP 주소 127.0.0.1 로 매핑해야 합니다 . • Windows 95 나 98 에서 이 파일은 \Windows\Hosts 에 있습니다 . • Windows NT4 에서 이 파일은 \WinNT\System32\drivers\etc\Hosts 에 있습니다 . 항목은 다음과 같습니다 . 127.0.0.1 exchange exchange.company.com Exchange 서버는 자체 이름을 Outlook 클라이언트로 다시 보냅니다 . 이러한 매핑을 통해 Outlook 클라이언트는 Netlet 클라이언트를 통해 서버에 다시 연결할 수 있게 됩니다 .

표 5-3 예제 Netlet 규칙

규칙	URL	애플릿 다운로드	로컬 포트	대상 호스트	대상 포트	설명
FTP	null	false	30021	*your-ftp_server.your-domain*	21	제어된 최종 사용자 계정과 함께 단일 FTP Server 에 FTP 서비스를 제공할 수 있습니다 . 그러면 최종 사용자 시스템에서 단일 위치로 보안 원격 FTP 전송이 이루어집니다 . 아이디가 없으면 FTP URL 은 익명의 FTP 연결로 해석됩니다 . 포트 30021 은 Netlet FTP 규칙에 사용할 로컬 포트로 정의해야 *합니다* . 동적 FTP 는 Netlet 연결을 통해 지원되지 않습니다 .
Netscape 4.7 우편 클라이언트	null	false	30143, 30025.	TARGET TARGET	10143 10025	Netscape 클라이언트에서 사용자는 다음을 지정해야 합니다 . IMAP 또는 수신 우편용 `localhost:30143` SMTP 또는 송신 우편용 `localhost:30025`
Graphon	third_party/xsession_start.html	true	10491	TARGET	491	Netlet 을 통해 Graphon 에 액세스하는 데 사용되는 규칙입니다 . `xsession_start.html` 은 Graphon 과 함께 번들로 제공됩니다 .
Citrix	third_party/citrix_start.html	true	1494	TARGET	1494	Netlet 을 통해 Citrix 에 액세스하는 데 사용되는 규칙입니다 . `citrix_start.html` 은 Citrix 와 함께 번들로 제공됩니다 .
원격 제어	third_party/pca_start.html	true	5631 5632	TARGET TARGET	5631 5632	Netlet 을 통해 원격 제어에 액세스하는 데 사용되는 규칙입니다 . `pca_start.html` 은 원격 제어와 함께 번들로 제공됩니다 .

Netlet 로깅 사용 설정

게이트웨이 서비스에서 Netlet 관련 활동의 로깅을 사용 설정할 수 있습니다 . 284 페이지의 "Netlet 기록 사용 " 을 참조하십시오 . 로그 파일은 Identity 서버 구성 속성의 로깅 섹션에 있는 로그 위치 속성에 지정된 디렉토리에 만들어집니다 .

로그 파일 이름에는 다음과 같은 규칙이 있습니다 .

srapNetlet_*gateway-hostname_gateway-profile-name*

Netlet 로그는 다음 정보를 캡처합니다 .

- 시작 시간
- 소스 주소
- 소스 포트
- 서버 주소
- 서버 포트
- 중단 시간
- 상태 (시작 또는 중단)

Sun Ray 환경에서 Netlet 실행

Sun Ray 환경에 있는 클라이언트 컴퓨터에 애플릿을 다운로드해야 하는 응용프로그램을 실행하려면 HTML 파일을 변경해야 합니다 . 다음은 필요한 수정 사항을 보여주는 예제 파일입니다 .

새로운 HTML 파일

```
<!-- @(#)citrix_start.html 2.1   98/08/17 Copyright (c) 1998 i-Planet, Inc., All rights
reserved.   -->

<html>

<script language="JavaScript">

var KEY_VALUES; // KEY_VALUES['key'] = 'value';

function retrieveKeyValues() {

      KEY_VALUES = new Object();

      var queryString = '' + this.location;

      queryString = unescape(queryString);

      queryString = queryString.substring((queryString.indexOf('?'))  + 1);

      if (queryString.length < 1) {
```

```
       return false; }
      var keypairs = new Object();
      var numKP = 0;
      while (queryString.indexOf('&') > -1) {
        keypairs[numKP] = queryString.substring(0,queryString.indexOf('&'));
        queryString = queryString.substring((queryString.indexOf('&')) + 1);
        numKP++;
      }
      // Store what's left in the query string as the final keypairs[] data.
      keypairs[numKP++] = queryString;
      var keyName;
      var keyValue;
      for (var i=0; i < numKP; ++i) {
        keyName = keypairs[i].substring(0,keypairs[i].indexOf('='));
        keyValue = keypairs[i].substring((keypairs[i].indexOf('=')) + 1);
        while (keyValue.indexOf('+') > -1) {
          keyValue = keyValue.substring(0,keyValue.indexOf('+')) + ' ' +
keyValue.substring(keyValue.indexOf('+') + 1);
        }
        keyValue = unescape(keyValue);
          // Unescape non-alphanumerics
        KEY_VALUES[keyName] = keyValue;
      }
}
function getClientPort(serverPort) {
    var keyName = "clientPort['" + serverPort +"']";
    return KEY_VALUES[keyName];
}
function generateContent() {
    retrieveKeyValues();
```

```
    var newContent =
        "<html>\n"
       + "<head></head>\n"
       + "<body>\n"
       + "<applet code=\"com.citrix.JICA.class\" archive=\"JICAEngN.jar\" width=800
height=600>\n"
       + "<param name=\"cabbase\" value=\"JICAEngM.cab\">\n"
       + "<param name=\"address\" value=\"localhost\">\n"
       + "<param name=ICAPortNumber value="
       + getClientPort('1494')
       + ">\n"
       + "</applet>\n"
       + "</body>\n"
       + "</html>\n";
    document.write(newContent);
}
</script>
<body onLoad="generateContent();">
</body>
</html>
```

Deprecated HTML 파일 :

```
<html>
<body>
<applet code="com.citrix.JICA.class" archive="JICAEngN.jar" width=800 height=600>
<param name="cabbase" value="JICAEngM.cab">
<param name="address" value="localhost">
<param name=ICAPortNumber value=1494>
</applet>
</body></html>
```

PDC 가 있는 Netlet

이 장에서는 Netlet 을 PDC 와 함께 사용할 수 있도록 클라이언트 브라우저의 Java 플러그인을 구성하는 방법을 설명합니다 .

참고 JSSE 가 있는 Virtual Machine (VM) 만 PDC 가 있는 Netlet 을 지원합니다 .

PDC 를 위한 Netlet 구성

➤ **PDC 를 위한 Netlet 을 구성하려면**

1. 다음 형식 중 하나로 브라우저에서 클라이언트 인증서를 내보냅니다 .
   - PKCS
   - JKS

   클라이언트 인증서를 내보낸 후 VM 에서 인증서를 사용하려면 Java 플러그인에 다음 JVM 매개 변수가 있어야 합니다 .

   javax.net.ssl.keyStoreType

   javax.net.ssl.keyStorePassword

   javax.netl.ssl.keyStore
2. [제어판] 으로 가서 Java 플러그인을 시작합니다 .
3. [고급] 탭을 선택하여 Java Runtime Environment 를 엽니다 .

4. Java 런타임 매개 변수를 지정합니다 . 예 :

   Djavax.net.ssl.keyStoreType=pkcs

   Djavax.net.ssl.keyStorePassword=testing123

   Djavax.netl.ssl.keyStore="C:\dir\test.cert"

5. [적용] 을 누릅니다 .
6. Java 플러그인을 닫고 관련 브라우저를 다시 시작합니다 .

7 장

인증서

이 장에서는 인증서 관리를 설명하고 직접 서명한 인증서와 인증 기관에서 받은 인증서를 설치하는 방법을 알아봅니다 .

이번 장에서는 다음 주제를 다룹니다 .

- SSL 인증서의 개요
- 인증서 파일
- 인증서 트러스트 속성
- CA 트러스트 속성
- certadmin 스크립트
- 직접 서명한 인증서 생성
- 인증 기관으로부터 SSL 인증서 설치
- 루트 CA 인증서 추가
- 인증서의 트러스트 속성 수정
- 루트 CA 인증서 나열
- 모든 인증서 나열
- 인증서 삭제
- 인증서 인쇄

SSL 인증서의 개요

Portal Server Secure Remote Access 소프트웨어는 인증서를 기반으로 원격 사용자를 인증합니다 . SRA 는 SSL (Secure Sockets Layer) 을 사용하여 안전한 통신을 가능하게 합니다 . SSL 프로토콜을 두 컴퓨터 간 보안 통신을 가능하도록 해줍니다 .

SSL 인증서에서는 공개 키와 개인 키 쌍을 사용하여 암호화 및 비밀번호 해독 기능을 제공합니다 .

인증서 유형은 2 가지입니다 .

- 직접 서명한 인증서 (또는 루트 CA 인증서라고도 함)
- 인증 기관 (CA) 에서 발급한 인증서

기본적으로 게이트웨이를 설치할 때는 직접 서명한 인증서가 생성 및 설치됩니다 .

설치후 언제라도 인증서를 설치 , 습득 또는 교체할 수 있습니다 .

SRA 도 개인 디지털 인증서 (PDC) 를 통해 클라이언트 인증을 지원합니다 . PDC 는 SSL 클라이언트 인증으로 사용자를 인증하는 메커니즘입니다 . SSL 클라이언트 인증을 사용하면 SSL 핸드셰이크가 게이트웨이에서 종료됩니다 . 게이트웨이는 사용자의 PDC 를 추출하여 인증된 서버로 전달합니다 . 그러면 이 서버는 PDC 를 사용하여 사용자를 인증합니다 . 인증 체인과 함께 PDC 를 구성하려면 76 페이지의 " 인증 체이닝 사용 " 을 참조하십시오 .

SRA 에는 SSL 인증서를 관리하는 데 사용할 수 있는 `certadmin` 이라는 도구가 있습니다 . 216 페이지의 "certadmin 스크립트 " 를 참조하십시오 .

인증서 파일

파일과 연관 있는 인증서는 `/etc/opt/SUNWps/cert/default/gateway-profile-name` 에 있습니다 . 이 디렉토리에는 기본적으로 파일이 5 개 들어 있습니다 .

표 7-1 에는 파일과 파일에 대한 설명이 나열되어 있습니다 .

표 7-1 인증서 파일

파일 이름	유형	설명
`cert8.db`, `key3.db`, `secmod.db`	이진	인증서 , 키 및 암호화 모듈을 위한 데이터가 들어 있습니다 . `certadmin` 스크립트로 조작할 수 있습니다 . Sun Java™ System 웹 서버에서 사용되는 데이터 파일과 형식이 같으며 *portal-server-install-root*`/SUNWwbsvr/alias` 에 있습니다 . 필요에 따라 이 파일은 Portal Server 호스트와 게이트웨이 구성 요소 또는 게이트웨이 사이에서 공유될 수 있습니다 .
`.jsspass`	숨은 텍스트 파일	SRA 키 데이터베이스를 위한 암호화된 비밀번호가 들어 있습니다 .
`.nickname`	숨은 텍스트 파일	*token-name:certificate-name* 형식으로 게이트웨이에서 사용해야 하는 토큰과 인증서 이름을 저장합니다 . 기본 토큰 (기본 내부 소프트웨어 암호화 모듈에 있는 토큰) 을 사용하는 경우 토큰 이름을 생략하십시오 . 대부분의 경우 `.nickname` 파일은 인증서 이름만 저장합니다 . 관리자로서 이 파일의 인증서 이름을 수정할 수 있습니다 . 지정한 인증서를 이제 게이트웨이에서 사용합니다 .

인증서 트러스트 속성

인증서의 트러스트 속성은 다음과 같은 정보를 나타냅니다 .

- 인증서 (클라이언트 또는 서버 인증서의 경우) 를 인증된 기관에서 발행했는지 여부 .
- 인증서 (루트 인증서의 경우) 가 서버 또는 클라이언트 인증서의 발급자로 인증될 수 있는지 여부 .

각 인증서에는 "SSL, 전자 메일 , 객체 서명 " 순서로 사용할 수 있는 트러스트 범주가 3 가지 있습니다 . 첫 번째 범주만 게이트웨이에 유용합니다 . 각 범주 위치에서 트러스트 속성 코드가 사용되지 않을 수도 있고 많이 사용되기도 합니다 .

범주에 대한 속성 코드는 쉼표로 분리되며 전체 속성 집합은 따옴표로 묶입니다 . 예를 들어 , 게이트웨이 설치 시 생성 및 설치된 직접 서명한 인증서는 "u,u,u" 로 표시되는 데 이는 루트 CA 인증서와는 반대로 서버 인증서 (사용자 인증서) 임을 의미합니다 .

표 7-2 에는 가능한 속성 값과 각 값의 의미가 나열되어 있습니다 .

표 7-2 인증서 트러스트 속성

속성	설명
p	유효한 피어
P	인증된 피어 (p 내포)
c	유효한 CA
T	클라이언트 인증서를 발급할 수 있도록 인증된 CA(c 내포)
C	서버 인증서를 발급할 수 있도록 인증된 CA(SSL 전용)(c 내포)
u	인증서를 인증이나 서명에 사용할 수 있음
w	경고 전송 (해당 컨텍스트에서 인증서가 사용될 경우 다른 속성과 함께 사용하여 경고 포함)

CA 트러스트 속성

잘 알려진 공인 CA 는 대부분 인증서 데이터베이스에 들어 있습니다 . 공인 CA 의 트러스트 속성을 수정하는 데 대한 내용은 226 페이지의 " 인증서의 트러스트 속성 수정 " 을 참조하십시오 .

표 7-3 에는 트러스트 속성이 있는 가장 일반적인 인증 기관이 나열되어 있습니다 .

표 7-3 공인 인증 기관

인증 기관 이름	트러스트 속성
Verisign/RSA Secure Server CA	CPp,CPp,CPp
VeriSign Class 4 Primary CA	CPp,CPp,CPp
GTE CyberTrust Root CA	CPp,CPp,CPp
GTE CyberTrust Global Root	CPp,CPp,CPp
GTE CyberTrust Root 5	CPp,CPp,CPp
GTE CyberTrust Japan Root CA	CPp,CPp,CPp
GTE CyberTrust Japan Secure Server CA	CPp,CPp,CPp

표 7-3 공인 인증 기관

Thawte Personal Basic CA	CPp,CPp,CPp
Thawte Personal Premium CA	CPp,CPp,CPp
Thawte Personal Freemail CA	CPp,CPp,CPp
Thawte Server CA	CPp,CPp,CPp
Thawte Premium Server CA	CPp,CPp,CPp
American Express CA	CPp,CPp,CPp
American Express Global CA	CPp,CPp,CPp
Equifax Premium CA	CPp,CPp,CPp
Equifax Secure CA	CPp,CPp,CPp
BelSign Object Publishing CA	CPp,CPp,CPp
BelSign Secure Server CA	CPp,CPp,CPp
TC TrustCenter, Germany, Class 0 CA	CPp,CPp,CPp
TC TrustCenter, Germany, Class 1 CA	CPp,CPp,CPp
TC TrustCenter, Germany, Class 2 CA	CPp,CPp,CPp
TC TrustCenter, Germany, Class 3 CA	CPp,CPp,CPp
TC TrustCenter, Germany, Class 4 CA	CPp,CPp,CPp
ABAecom (sub., Am. Bankers Assn.) Root CA	CPp,CPp,CPp
Digital Signature Trust Co. Global CA 1	CPp,CPp,CPp
Digital Signature Trust Co. Global CA 3	CPp,CPp,CPp
Digital Signature Trust Co. Global CA 2	CPp,CPp,CPp
Digital Signature Trust Co. Global CA 4	CPp,CPp,CPp
Deutsche Telekom AG Root CA	CPp,CPp,CPp
Verisign Class 1 Public Primary Certification Authority	CPp,CPp,CPp
Verisign Class 2 Public Primary Certification Authority	CPp,CPp,CPp
Verisign Class 3 Public Primary Certification Authority	CPp,CPp,CPp

표 7-3 공인 인증 기관

Verisign Class 1 Public Primary Certification Authority - G2	CPp,CPp,CPp
Verisign Class 2 Public Primary Certification Authority - G2	CPp,CPp,CPp
Verisign Class 3 Public Primary Certification Authority - G2	CPp,CPp,CPp
Verisign Class 4 Public Primary Certification Authority - G2	CPp,CPp,CPp
GlobalSign Root CA	CPp,CPp,CPp
GlobalSign Partners CA	CPp,CPp,CPp
GlobalSign Primary Class 1 CA	CPp,CPp,CPp
GlobalSign Primary Class 2 CA	CPp,CPp,CPp
GlobalSign Primary Class 3 CA	CPp,CPp,CPp
ValiCert Class 1 VA	CPp,CPp,CPp
ValiCert Class 2 VA	CPp,CPp,CPp
ValiCert Class 3 VA	CPp,CPp,CPp
Thawte Universal CA Root	CPp,CPp,CPp
Verisign Class 1 Public Primary Certification Authority - G3	CPp,CPp,CPp
Verisign Class 2 Public Primary Certification Authority - G3	CPp,CPp,CPp
Verisign Class 3 Public Primary Certification Authority - G3	CPp,CPp,CPp
Verisign Class 4 Public Primary Certification Authority - G3	CPp,CPp,CPp
Entrust.net Secure Server CA	CPp,CPp,CPp
Entrust.net Secure Personal CA	CPp,CPp,CPp
Entrust.net Premium 2048 Secure Server CA	CPp,CPp,CPp
ValiCert OCSP Responder	CPp,CPp,CPp
Baltimore CyberTrust Code Signing Root	CPp,CPp,CPp
Baltimore CyberTrust Root	CPp,CPp,CPp
Baltimore CyberTrust Mobile Commerce Root	CPp,CPp,CPp
Equifax Secure Global eBusiness CA	CPp,CPp,CPp

표 7-3 공인 인증 기관

Equifax Secure eBusiness CA 1	CPp,CPp,CPp
Equifax Secure eBusiness CA 2	CPp,CPp,CPp
Visa International Global Root 1	CPp,CPp,CPp
Visa International Global Root 2	CPp,CPp,CPp
Visa International Global Root 3	CPp,CPp,CPp
Visa International Global Root 4	CPp,CPp,CPp
Visa International Global Root 5	CPp,CPp,CPp
beTRUSTed Root CA	CPp,CPp,CPp
Xcert Root CA	CPp,CPp,CPp
Xcert Root CA 1024	CPp,CPp,CPp
Xcert Root CA v1	CPp,CPp,CPp
Xcert Root CA v1 1024	CPp,CPp,CPp
Xcert EZ	CPp,CPp,CPp
CertEngine CA	CPp,CPp,CPp
BankEngine CA	CPp,CPp,CPp
FortEngine CA	CPp,CPp,CPp
MailEngine CA	CPp,CPp,CPp
TraderEngine CA	CPp,CPp,CPp
USPS Root	CPp,CPp,CPp
USPS Production 1	CPp,CPp,CPp
AddTrust Non-Validated Services Root	CPp,CPp,CPp
AddTrust External Root	CPp,CPp,CPp
AddTrust Public Services Root	CPp,CPp,CPp
AddTrust Qualified Certificates Root	CPp,CPp,CPp
Verisign Class 1 Public Primary OCSP Responder	CPp,CPp,CPp

표 7-3 공인 인증 기관

Verisign Class 2 Public Primary OCSP Responder	CPp,CPp,CPp
Verisign Class 3 Public Primary OCSP Responder	CPp,CPp,CPp
Verisign Secure Server OCSP Responder	CPp,CPp,CPp
Verisign Time Stamping Authority CA	CPp,CPp,CPp
Thawte Time Stamping CA	CPp,CPp,CPp
E-Certify CA	CPp,CPp,CPp
E-Certify RA	CPp,CPp,CPp
Entrust.net Global Secure Server CA	CPp,CPp,CPp
Entrust.net Global Secure Personal CA	CPp,CPp,CPp

certadmin 스크립트

다음과 같은 인증서 관리 작업에 `certadmin` 스크립트를 사용할 수 있습니다 .

- 직접 서명한 인증서 생성
- 인증서 서명 요청 (CSR) 생성
- 루트 CA 인증서 추가
- CA 로부터 받은 인증서 설치
- 인증서 삭제
- 인증서의 트러스트 속성 수정
- 루트 CA 인증서 나열
- 모든 인증서 나열
- 인증서 인쇄

직접 서명한 인증서 생성

각 서버와 게이트웨이 사이의 SSL 통신을 위해서는 인증서를 생성해야 합니다 .

➤ 설치 후 직접 서명한 인증서를 생성하려면

1. 루트로서 인증서를 생성하고자 하는 게이트웨이 컴퓨터에 `certadmin` 스크립트를 실행합니다 .

   portal-server-install-root`/SUNWps/bin/certadmin -n` *gateway-profile-name*

   인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates

9) Print Certificate Content

10) Quit

choice: [10] 1
```

2. 인증서 관리 메뉴의 옵션 **1** 를 선택합니다 .

   인증서 관리 스크립트에서 기존 데이터베이스 파일을 유지할 것인지 묻습니다 .

3. 조직별 정보 , 토큰 이름 및 인증서 이름을 입력합니다 .

참고 와일드카드 인증서에는 호스트의 완전한 정규 DNS 이름에 * 를 지정합니다 . 예를 들어 , 호스트의 완전한 정규 DNS 이름이 `abc.sesta.com` 이면 `*.sesta.com` 으로 지정합니다 . 이제 생성된 인증서는 `sesta.com` 도메인에 있는 모든 호스트 이름에 유효합니다 .

```
What is the fully-qualified DNS name of this host?  [host_name.domain_name]

What is the name of your organization (ex: Company)?  []

What is the name of your organizational unit (ex: division)?  []

What is the name of your City or Locality?  []

What is the name (no abbreviation please) of your State or Province?  []

What is the two-letter country code for this unit?  []

Token name is needed only if you are not using the default internal
(software) cryptographic module, for example, if you want to use a crypto
card (Token names could be listed using: modutil -dbdir
/etc/opt/SUNWps/cert/gateway-profile-name -list); Otherwise, just hit Return
below.

Please enter the token name.  []

Enter the name you like for this certificate?

Enter the validity period for the certificate (months) [6]
A self-signed certificate is generated and the prompt returns.
```

토큰 이름 (기본적으로는 비어 있음) 과 인증서 이름은 `/etc/opt/SUNWps/cert/`*gateway-profile-name* 에 있는 `.nickname` 파일에 저장됩니다 .

4. 인증서가 적용되도록 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *new gateway-profile-name* start

인증서 서명 요청 (CSR) 생성

CA 로부터 인증서를 주문하기 전에 CA 에서 요구하는 정보가 들어 있는 인증서 서명 요청을 만들어야 합니다 .

➤ CSR 을 생성하려면

1. 루트로서 certadmin 스크립트를 실행합니다 .

   portal-server-install-root/SUNWps/bin/certadmin -n *gateway-profile-name*

   인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates

9) Print Certificate Content

10) Quit

choice: [10] 2
```

2. 인증서 관리 메뉴의 옵션 **2** 를 선택합니다 .

스크립트에서 조직별 정보 , 토큰 이름 및 웹 마스터의 전자 우편과 전화 번호를 입력하라는 메시지를 표시합니다 .

호스트의 완전한 정규 DNS 이름을 반드시 지정해야 합니다 .

```
What is the fully-qualified DNS name of this host?  [snape.sesta.com]

What is the name of your organization (ex: Company)?  []

What is the name of your organizational unit (ex: division)?  []

What is the name of your City or Locality?  []

What is the name (no abbreviation please) of your State or Province?  []

What is the two-letter country code for this unit?  []

Token name is needed only if you are not using the default internal
(software) cryptographic module, for example, if you want to use a crypto
card (Token names could be listed using: modutil -dbdir /etc/opt/SUNWps/cert
-list); Otherwise, just hit Return below.

Please enter the token name []

Now input some contact information for the webmaster of the machine that the
certificate is to be generated for.

What is the email address of the admin/webmaster for this server [] ?

What is the phone number of the admin/webmaster for this server [] ?
```

3. 필요한 정보를 모두 입력하십시오 .

참고 웹 마스터의 전자 우편과 전화 번호를 공백으로 남겨두지 마십시오 . 이 정보는 유효한 CSR 을 받는 데 필요합니다 .

CSR 이 생성되어
portal-server-install-root/SUNWps/bin/csr.hostname.datetimestamp 파일에 저장됩니다 . CSR 은 화면에도 인쇄됩니다 . CA 로부터 인증서를 주문할 때 CSR 을 직접 복사한 후 붙여넣을 수 있습니다 .

루트 CA 인증서 추가

클라이언트 사이트에서 게이트웨이 인증서 데이터베이스에 알려지지 않은 CA 에서 서명한 인증서를 제시하면 SSL 핸드셰이크가 실패합니다 .

이를 방지하려면 루트 CA 인증서를 인증서 데이터베이스에 추가해야 합니다 . 그러면 게이트웨이에서 CA 를 인식할 수 있게 됩니다 .

CA 의 웹 사이트를 찾아서 해당 CA 의 루트 인증서를 얻으십시오 . certadmin 스크립트를 사용할 때 파일 이름과 루트 CA 인증서의 경로를 지정합니다 .

➤ 루트 CA 인증서를 추가하려면

1. 루트로서 certadmin 스크립트를 실행합니다 .

   portal-server-install-root/SUNWps/bin/certadmin -n *gateway-profile-name*

   인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates
```

```
9) Print Certificate Content

10) Quit

choice: [10] 3
```

2. 인증서 관리 메뉴의 옵션 3 를 선택합니다 .
3. 루트 인증서가 들어 있는 파일 이름을 입력한 다음 인증서 이름을 입력합니다 .

   그러면 루트 CA 인증서가 인증서 데이터베이스에 추가됩니다 .

인증 기관으로부터 SSL 인증서 설치

게이트웨이를 설치하는 동안 기본적으로 직접 서명한 인증서가 만들어져 설치됩니다 . 설치 후 언제라도 공식 인증 기관 (CA) 서비스를 제공하는 공급업체나 기업 CA 에 의해 서명된 SSL 인증서를 설치할 수 있습니다 .

이 작업은 다음과 같은 3 단계로 이루어집니다 .

- 인증서 서명 요청 (CSR) 생성
- CA 로부터 인증서 주문
- CA 로부터 받은 인증서 설치

CA 로부터 인증서 주문

인증서 서명 요청 (CSR) 을 만들었으면 CSR 을 사용하여 CA 로부터 인증서를 주문해야 합니다 .

➤ **CA 로부터 인증서를 주문하려면**

1. 인증 기관의 웹 사이트로 가서 인증서를 주문합니다 .

2. CA 의 요청에 따라 CSR 을 제공합니다 . CA 의 요청에 따라 기타 정보도 제공합니다 .

   그러면 CA 로부터 인증서를 받게 됩니다 . 인증서를 파일에 저장합니다 . 파일에 인증서와 함께 "BEGIN CERTIFICATE" 및 "END CERTIFICATE" 라인을 포함시킵니다 .

   다음 예제에서는 실제 인증서 데이터를 생략하였습니다 .

```
-----BEGIN CERTIFICATE-----

The certificate contents...

----END CERTIFICATE-----
```

CA 로부터 받은 인증서 설치

certadmin 스크립트를 사용하여 CA 로부터 받은 인증서를 /etc/opt/SUNWps/cert/*gateway-profile-name* 의 로컬 데이터베이스 파일에 설치합니다 .

➤ CA 로부터 받은 인증서를 설치하려면

1. 루트로서 certadmin 스크립트를 실행합니다 .

   portal-server-install-root/SUNWps/bin/certadmin -n *gateway-profile-name*

   인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate
```

```
6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates

9) Print Certificate Content

10) Quit

choice: [10] 4
```

2. 인증서 관리 메뉴의 옵션 **4** 를 선택합니다 .

   스크립트에서 인증서 파일 이름 , 인증서 이름 및 토큰 이름을 입력하라고 요청합니다 .

```
What is the name (including path) of file that contains the certificate?
Please enter the token name you used when creating CSR for this certificate.
[]
```

3. 필요한 정보를 모두 입력하십시오 .

   인증서가 `/etc/opt/SUNWps/cert/`*gateway-profile-name* 에 설치되고 화면 메시지가 나타납니다 .

4. 인증서가 적용되도록 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

인증서 삭제

인증서 관리 스크립트를 사용하면 인증서를 삭제할 수 있습니다 .

➤ 인증서를 삭제하려면

1. 루트로서 `certadmin` 스크립트를 실행합니다 .

```
portal-server-install-root/SUNWps/bin/certadmin -n gateway-profile-name
```

여기서 *gateway-profile-name* 은 게이트웨이 인스턴스의 이름입니다 .

인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates

9) Print Certificate Content

10) Quit

choice: [10] 5
```

2. 인증서 관리 메뉴의 옵션 **5** 를 선택합니다 .

3. 삭제할 인증서의 이름을 입력하십시오 .

인증서의 트러스트 속성 수정

인증서의 트러스트 속성을 수정해야 하는 한 경우는 게이트웨이에서 클라이언트 인증이 사용될 때입니다 . 클라이언트 인증의 한 예는 PDC(Personal Digital Certificate) 입니다 . PDC 를 발급하는 CA 는 게이트웨이에 의해 인증되어야 하며 CA 인증서에는 SSL 용으로 "T" 라고 표시되어 있어야 합니다 .

게이트웨이 구성 요소가 HTTPS 사이트와 통신하도록 설정된 경우 HTTPS 사이트 서버 인증서의 CA 는 게이트웨이에서 인증되어야 하며 CA 인증서에는 SSL 용의 "C" 표시가 있어야 합니다 .

➤ 인증서의 트러스트 속성을 수정하려면

1. 루트로서 `certadmin` 스크립트를 실행합니다 .

```
gateway-install-root/SUNWps/bin/certadmin -n gateway-profile-name
```

여기서 *gateway-profile-name* 은 게이트웨이 인스턴스의 이름입니다 .

인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates
```

```
9) Print Certificate Content

10) Quit

choice: [10] 6
```

2. 인증서 관리 메뉴의 옵션 6 를 선택합니다 .
3. 인증서의 이름을 입력합니다 . 예를 들어 , Thawte Personal Freemail C 와 같이 입력하면 됩니다 .

```
Please enter the name of the certificate?
Thawte Personal Freemail CA
```

4. 인증서의 트러스트 속성을 입력합니다 .

```
Please enter the trust attribute you want the certificate to have [CT,CT,CT]
```

인증서 트러스트 속성이 변경됩니다 .

루트 CA 인증서 나열

인증서 관리 스크립트를 사용하면 모든 루트 CA 인증서를 볼 수 있습니다 .

➤ **루트 CA 목록을 보려면**

1. 루트로서 certadmin 스크립트를 실행합니다 .

```
portal-server-install-root/SUNWps/bin/certadmin -n gateway-profile-name
```

여기서 *gateway-profile-name* 은 게이트웨이 인스턴스의 이름입니다 .

인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates

9) Print Certificate Content

10) Quit

choice: [10] 7
```

2. 인증서 관리 메뉴의 옵션 **7** 를 선택합니다 .

모든 루트 CA 인증서가 표시됩니다 .

모든 인증서 나열

인증서 관리 스크립트를 사용하면 모든 인증서와 상응하는 트러스트 속성을 볼 수 있습니다 .

➤ **모든 인증서를 나열하려면**

1. 루트로서 certadmin 스크립트를 실행합니다 .

```
portal-server-install-root/SUNWps/bin/certadmin -n gateway-profile-name
```

여기서 *gateway-profile-name* 은 게이트웨이 인스턴스의 이름입니다 .

인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates

9) Print Certificate Content

10) Quit

choice: [10] 8
```

2. 인증서 관리 메뉴의 옵션 **8** 를 선택합니다 .

   모든 CA 인증서가 표시됩니다 .

인증서 인쇄

인증서 관리 스크립트를 사용하면 인증서를 인쇄할 수 있습니다 .

➤ **인증서를 인쇄하려면**

1. 루트로서 `certadmin` 스크립트를 실행합니다 .

```
portal-server-install-root/SUNWps/bin/certadmin -n gateway-profile-name
```

   여기서 *gateway-profile-name* 은 게이트웨이 인스턴스의 이름입니다 .

   인증서 관리 메뉴가 표시됩니다 .

```
1) Generate Self-Signed Certificate

2) Generate Certificate Signing Request (CSR)

3) Add Root CA Certificate

4) Install Certificate From Certificate Authority (CA)

5) Delete Certificate

6) Modify Trust Attributes of Certificate (e.g., for PDC)

7) List Root CA Certificates

8) List All Certificates
```

```
9) Print Certificate Content

10) Quit

choice: [10] 9
```

2. 인증서 관리 메뉴의 옵션 **9** 를 선택합니다 .
3. 인증서의 이름을 입력합니다 .

URL 액세스 제어 구성

이 장에서는 특정 URL 의 게이트웨이를 통해 [SRA 구성], [액세스 목록] 의 Sun Java™ System Identity Server 관리 콘솔에서 최종 사용자에게 액세스를 허용하거나 거부하는 방법을 설명합니다 .

참고 Identity Server 관리 콘솔의 맨 위 오른쪽 구석에서 [설명서] 를 누르고 [SRA 도움말] 을 누르면 모든 Sun Java System Portal Server, Secure Remote Access (SRA) 속성을 빠르게 참조할 수 있습니다 .

➤ **URL 액세스 제어를 구성하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. 관리 콘솔에서 [서비스 구성] 탭을 선택합니다 .
3. [SRA 구성] 에서 [액세스 목록] 옆에 있는 화살표를 클릭합니다 .

   액세스 목록 페이지가 나타납니다 .

이 페이지에서는 다음 작업을 수행할 수 있습니다 .

- 거부된 URL 목록 설정
- 허용된 URL 목록 설정
- 단일 사인온 관리

참고 SRA 을 설치할 때 모든 사용자가 기본적으로 액세스 목록 서비스를 사용할 수 있는 것은 아닙니다 . 이 서비스는 설치 시 기본적으로 만들어지는 amadmin 사용자에게만 사용 설정되어 있습니다 . 다른 사용자는 이 서비스를 사용하지 않고 게이트웨이를 통해 데스크탑에 액세스할 수 없습니다 . amadmin 로 로그인하여 이 서비스를 모든 사용자에게 할당합니다 .

거부된 URL 목록 설정

이 필드에서 최종 사용자가 게이트웨이를 통해 액세스할 수 없는 URL 목록을 지정할 수 있습니다 .

게이트웨이에서는 [허용된 URL] 목록을 확인하기 전에 [거부된 URL] 목록을 확인합니다 .

➤ **거부된 URL 목록을 설정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. [SRA 구성] 에서 [액세스 목록] 옆에 있는 화살표를 클릭합니다 .

   액세스 목록 페이지가 나타납니다 .
4. [거부된 URL] 필드에서 게이트웨이를 통해 액세스를 거부할 URL 을 지정합니다 . URL 입력 형식 :

   `http://abc.siroe.com`
5. [추가] 를 누릅니다 .

   그러면 해당 URL 이 [거부된 URL] 목록에 추가됩니다 .

   `http://*.siroe.com` 과 같은 일반 식을 사용할 수도 있습니다 . 이 경우 사용자는 `siroe.com` 도메인에 있는 모든 호스트에 액세스가 거부됩니다 .
6. [저장] 을 눌러 변경 사항을 기록합니다 .

허용된 URL 목록 설정

게이트웨이를 통해 최종 사용자가 액세스할 수 있는 모든 URL 를 지정할 수 있습니다 . 기본적으로 이 목록에는 와일드카드 항목 (*) 이 있으므로 모든 URL 에 액세스할 수 있다는 것을 의미합니다 . 모든 URL 에 대한 액세스를 허용하고 , 특정 URL 에 대한 액세스만 제한하려면 제한되는 URL 을 [거부된 URL] 목록에 추가합니다 . 같은 방식으로 특정 URL 의 액세스만 허용하려면 [거부된 URL] 필드를 비워두고 [허용된 URL] 필드에서 필요한 URL 을 지정합니다 .

게이트웨이에서는 [허용된 URL] 을 확인하기 전에 [거부된 URL] 을 확인합니다 .

➤ **허용된 URL 목록을 설정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. [SRA 구성] 에서 [액세스 목록] 옆에 있는 화살표를 클릭합니다 .

   액세스 목록 페이지가 나타납니다 .
4. [허용된 URL] 필드에서 게이트웨이를 통해 액세스를 허용할 URL 을 지정합니다 . URL 입력 형식 :

   ```
   http://abc.siroe.com
   ```
5. [추가] 를 누릅니다 .

   그러면 해당 URL 이 [허용된 URL] 에 추가됩니다 .

참고 [허용된 URL] 필드에는 기본적으로 * 가 있으며 , 따라서 게이트웨이를 통해 모든 URL 에 액세스할 수 있습니다 .

6. [저장] 을 눌러 변경 사항을 기록합니다

단일 사인온 관리

SRA 소프트웨어의 액세스 목록 서비스에서는 다양한 호스트에 대한 단일 사인온 기능을 제어할 수 있습니다 . 단일 사인온 기능을 사용하려면 게이트웨이 서비스에서 [HTTP 기본 인증 사용] 옵션을 활성화해야 합니다 . 241 페이지의 "HTTP 및 HTTPS 연결 사용 " 을 참조하십시오 .

액세스 목록 서비스에서는 특정 호스트에 대한 단일 사인온을 사용 해제할 수 있습니다 . 이것은 각 세션마다 단일 사인온이 사용 설정되어 있지 않는 한 , HTPP 기본 인증이 필요한 호스트에 연결할 때마다 최종 사용자가 인증 받아야 한다는 것을 의미합니다 .

특정 호스트에 단일 사인온을 사용 해제해 놓은 경우 사용자는 단일 Portal Server 세션 내에서 만 그 호스트에 연결할 수 있습니다 . 예를 들어 , `abc.sesta.com` 에 단일 사인온을 사용 불가능으로 설정했다고 가정해 봅시다 . 사용자가 이 사이트에 처음 연결할 때는 인증이 필요합니다 . 사용자는 다른 페이지를 찾아본 후 나중에 이 페이지로 돌아올 수 있는데 페이지가 같은 Portal Server 세션 내에 있으면 인증이 필요치 않습니다 .

사용자는 제한 관리 콘솔을 사용해서 이 등록 정보를 구성할 수도 있습니다 .

➤ 호스트의 단일 사인온을 비활성화하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. [SRA 구성] 에서 [액세스 목록] 옆에 있는 화살표를 클릭합니다 .

   액세스 목록 페이지가 나타납니다 .
4. [단일 사인온이 금지된 호스트] 필드에서 단일 사인온 (SSO) 을 비활성화할 호스트를 지정합니다 .

   `abc.siroe.com` 과 같은 형식으로 호스트 이름을 지정합니다 .
5. [추가] 를 누릅니다 .

   해당 호스트 이름이 목록에 추가됩니다 .
6. [저장] 을 눌러 변경 사항을 기록합니다

➤ 세션별 단일 사인온을 활성화하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. [SRA 구성] 에서 [액세스 목록] 옆에 있는 화살표를 클릭합니다 .

   액세스 목록 페이지가 나타납니다 .
4. [세션별 단일 사인온 허용] 확인란을 선택하여 세션의 단일 사인온을 활성화합니다 .
5. [저장] 을 눌러 변경 사항을 기록합니다 .

➤ 인증 수준을 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. [SRA 구성] 에서 [액세스 목록] 옆에 있는 화살표를 클릭합니다 .

   액세스 목록 페이지가 나타납니다 .
4. [허용된 인증 수준] 필드로 스크롤합니다 .
5. 허용되는 인증을 입력합니다 . 모든 수준을 허용하려면 별표를 사용합니다 .
6. [저장] 을 눌러 변경 사항을 기록합니다 .

게이트웨이 구성

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 Gateway 속성을 구성하는 방법에 대해 설명합니다 .

참고 Identity Server 관리 콘솔의 맨 위 오른쪽 구석에서 [도움말] 을 누르고 [SRA 도움말] 을 누르면 모든 Sun Java System Portal Server Secure Remote Access (SRA) 속성을 빠르게 참조할 수 있습니다 .

게이트웨이를 설정하려면 38 페이지의 " 게이트웨이 프로필 만들기 " 를 참조하십시오 .

게이트웨이 프로필을 만든 다음에는 게이트웨이 속성을 구성해야 합니다 .

➤ 게이트웨이 속성을 구성하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. 관리 콘솔에서 [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

   여기서 해당 탭을 클릭합니다 .

   - 핵심 탭
   - 프록시 탭
   - 보안 탭

- Rewriter 탭
- 기록 탭

각 탭에서 구성할 수 있는 탭과 속성을 아래에서 설명합니다 .

핵심 탭

게이트웨이 서비스의 [핵심] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- HTTP 및 HTTPS 연결 사용
- Rewriter 프록시 목록 사용과 만들기
- Netlet 활성화
- Netlet 프록시 목록 활성화 및 만들기
- Proxylet 활성화
- 쿠키 관리 활성화
- HTTP 기본 인증 사용
- HTTP 지속 연결 사용
- 지속 연결당 최대 요청 수 지정
- 지속성 소켓 연결에 시간 초과 지정
- 반환 시간을 위한 계정의 유예 시간 초과 지정
- 사용자 세션 쿠키를 URL 로 전달 목록 만들기
- 최대 연결 대기 길이 지정
- 게이트웨이 시간 초과 지정
- 최대 스레드 풀 크기 지정
- 캐시된 소켓 시간 초과 지정
- Portal Server 목록 만들기
- 서버 재시도 간격 지정
- 외부 서버 쿠키 저장 활성화
- URL 에서 세션 얻기
- 쿠키를 안전하다고 표시

HTTP 및 HTTPS 연결 사용

설치 중에 HTTPS 모드에서 게이트웨이를 실행하도록 선택한 경우 설치 후에 게이트웨이가 HTTPS 모드에서 실행됩니다 . HTTPS 모드에서 게이트웨이는 브라우저로부터 SSL 연결을 허용하고 SSL 이 아닌 연결은 거부합니다 .

그러나 , 게이트웨이를 HTTP 모드에서 실행되도록 설정할 수도 있습니다 . 그러면 SSL 세션 관리와 SSL 트래픽 암호화 및 해독에 관련된 오버헤드가 생기지 않기 때문에 게이트웨이 성능을 높일 수 있습니다 .

➤ **HTTP 또는 HTTPS 모드에서 실행되도록 게이트웨이를 구성하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. 관리 콘솔에서 [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭에서 다음을 수행합니다 .
   - 필요에 따라 [HTTP 연결 사용], [HTTPS 연결 사용] 또는 두 확인란을 모두 선택합니다 .
   - [HTTPS 포트] 필드에서 필요한 HTTPS 포트를 지정합니다 .
   - [HTTP 포트] 필드에서 필요한 HTTP 포트를 지정합니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

Rewriter 프록시 목록 사용과 만들기

Rewriter 프록시는 게이트웨이와 인트라넷 컴퓨터 간에 안전한 HTTP 트래픽을 가능하게 합니다 . Rewriter 프록시를 지정하지 않으면 사용자가 인트라넷 컴퓨터에 액세스하려고 할 때 게이트웨이 구성 요소에서 인트라넷 컴퓨터에 직접 연결을 구성합니다 .

Rewriter 프록시는 설치 후에 자동으로 실행되지 않습니다 . 아래 설명에 따라 Rewriter 프록시를 활성화해야 합니다 .

➤ Rewriter 프록시를 사용하고 목록을 만들려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

참고 Rewriter 프록시와 게이트웨이가 같은 게이트웨이 프로필을 사용해야 합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [Rewriter 프록시 사용] 확인란을 선택하여 Rewriter 프록시를 활성화합니다 .
7. [Rewriter 프록시] 편집 상자에 필요한 호스트와 포트를 `hostname:port` 형식으로 입력합니다 .
8. [추가] 를 누릅니다 .
9. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
10. 서버에서 *portal-server-install-root*`/SUNWps/bin/certadmin` 을 실행하여 Rewriter 프록시의 인증서를 만듭니다 .

    Rewriter 프록시 설치 중에 인증서를 만들지 않은 경우에만 이 단계가 필요합니다 .
11. Rewriter 프록시가 설치된 컴퓨터에 루트로 로그인하여 Rewriter 프록시를 시작합니다 .

    rewriter-proxy-install-root`/SUNWps/bin/rwproxyd -n` *gateway-profile-name* `start`
12. 게이트웨이가 설치된 컴퓨터에 루트로 로그인하여 게이트웨이를 시작합니다 .

    gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

Netlet 활성화

Netlet 을 사용하여 사용자는 인터넷과 같은 불안전한 네트워크에서 공통 TCP/IP 서비스를 안전하게 실행할 수 있습니다 . TCP/IP 응용프로그램 (텔넷 및 SMTP), HTTP 응용프로그램 및 고정 포트 응용프로그램을 실행할 수 있습니다 .

Netlet 의 사용이 설정되면 게이트웨이에서 들어오는 트래픽이 Netlet 트래픽인지 또는 Portal Server 트래픽인지를 결정해야 합니다 . Netlet 을 사용 불가능으로 설정하면 게이트웨이가 들어오는 모든 트래픽이 HTTP 이거나 HTTPS 트래픽이라고 가정하기 때문에 이러한 오버헤드가 줄어듭니다 . Portal Server 에서 응용 프로그램을 전혀 사용하지 않을 Netlet 사용을 해제합니다 .

➤ **Netlet 을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [Netlet 사용] 확인란을 선택합니다 . 이 확인란은 기본적으로 선택됩니다 . 선택을 취소하면 Netlet 이 비활성화됩니다 .
7. [Netlet 프록시 사용] 확인란을 선택하여 Netlet 프록시를 활성화합니다 .
8. [Netlet 프록시 목록] 편집 상자에 필요한 호스트와 포트를 `hostname:port` 형식으로 입력합니다 .
9. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
10. 단말기 창에서 게이트웨이를 다시 시작합니다 .

    gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

Netlet 프록시 목록 활성화 및 만들기

Netlet 프록시는 인트라넷에 상주하는 Netlet 프록시로 가는 게이트웨이를 통해 클라이언트로부터의 보안 터널을 확장하여 게이트웨이와 인트라넷 사이에서 Netlet 트래픽의 보안을 강화합니다 .

Netlet 프록시가 활성화되면 Netlet 패킷은 Netlet 프록시에서 해독되어 대상 서버로 전달됩니다 . 이를 통해 방화벽에서 열어야 하는 포트 수가 줄어듭니다 .

➤ Netlet 프록시를 사용하고 목록을 만들려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. 왼쪽 프레임에서 SRA 구성 아래의 게이트웨이 옆에 있는 오른쪽 화살표를 클릭합니다 .

   오른쪽 창에 게이트웨이 페이지가 표시됩니다 .
4. 필요한 프로필 옆의 [편집] 을 클릭합니다 .

   오른쪽 창에 게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [Netlet 프록시 사용] 확인란을 선택하여 Netlet 프록시를 활성화합니다 .
6. [Netlet 프록시 호스트] 필드에 필요한 Netlet 프록시 호스트와 포트를 `host hostname:port` 형식으로 입력합니다 .

팁 필요한 포트가 있고 아직 사용되지 않았는지 확인하려면 명령줄에서 다음을 입력합니다 .

`netstat -a` | **`grep`** *port-number* | **`wc -l`**

port-number 가 필요한 포트입니다 .

---

7. [추가] 를 누릅니다 .
8. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
9. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

Proxylet 활성화

➤ **Proxylet 을 활성화하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .
7. [Proxylet 사용] 확인란을 선택합니다 .
8. [프록시] 탭을 눌러 [도메인 및 부속 도메인의 프록시] 필드로 스크롤해 내려간 후 게이트웨이로 연결되는 URL 의 도메인을 입력합니다 .
9. [저장] 을 누릅니다 .

쿠키 관리 활성화

많은 웹 사이트에서는 사용자 세션을 추적하고 관리하기 위해 쿠키를 사용합니다 . 게이트웨이가 HTTP 헤더에 쿠키를 설정하는 요청을 웹 사이트로 보낼 때 게이트웨이는 이러한 쿠키를 다음과 같이 폐기시키거나 통과시킵니다 .

- 게이트웨이 서비스에서 [쿠키 관리 사용] 속성이 선택되지 않으면 쿠키가 다시 작성되지 않습니다 . 따라서 브라우저로부터의 쿠키는 인트라넷에 도달하지 못합니다 (반대의 경우도 마찬가지).
- [쿠키 관리 사용] 속성이 선택되면 게이트웨이가 쿠키를 다시 작성합니다 . 게이트웨이는 브라우저로부터의 쿠키가 대상 인트라넷 호스트에 도달하도록 합니다 (반대의 경우도 마찬가지).

이 설정은 Portal Server 에서 Portal Server 사용자 세션 추적에 사용되는 쿠키에는 적용되지 않습니다 . [사용자 세션 쿠키가 전달될 사용자 세션 URL] 옵션의 구성으로 제어합니다 . 250 페이지의 " 사용자 세션 쿠키를 URL 로 전달 목록 만들기 " 를 참조하십시오 .

이 설정은 사용자가 액세스할 수 있는 모든 웹 사이트에 적용됩니다 (즉 , 어떤 사이트의 쿠키는 폐기시키고 어떤 사이트의 쿠키는 유지할 수 없습니다).

참고 쿠키 없는 게이트웨이에서라도 [쿠키 도메인] 목록에서 URL 을 제거하지 마십시오 . 쿠키 도메인 목록에 대한 내용은 *Identity Server 관리 설명서*를 참조하십시오 .

➤ 쿠키 관리를 사용하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
1. [서비스 구성] 탭을 선택합니다 .
2. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
3. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
4. [핵심] 탭을 클릭합니다 .
5. [쿠키 관리 사용] 확인란을 선택하고 쿠키 관리의 사용을 설정합니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

HTTP 기본 인증 사용

HTTP 기본 인증은 게이트웨이 서비스에서 설정할 수 있습니다 .

HTTP 기본 인증을 통해 방문자가 사이트를 보기 전에 아이디와 비밀번호를 입력하도록 요구함으로써 웹 사이트를 보호할 수 있습니다 (HTTP 응답 코드 401, WWW-인증서 :BASIC). Portal Server 는 사용자가 BASIC 으로 보호된 웹 사이트를 다시 방문할 때 자격 증명 정보를 다시 입력할 필요가 없도록 사용자 이름과 비밀번호를 저장할 수 있습니다 . 이러한 자격 증명 정보는 디렉토리 서버의 사용자 프로필에 저장됩니다 .

이 설정은 사용자가 BASIC 으로 보호된 사이트를 방문할 수 있는지 여부가 아니라 사용자가 입력한 자격 증명 정보를 사용자 프로필에 저장할지 여부만을 결정합니다.

이 설정은 사용자가 액세스할 수 있는 모든 웹 사이트에 적용됩니다 (즉 , HTTP 기본 인증 캐싱을 어떤 사이트에 사용하고 다른 사이트에 사용하지 않을 수는 없습니다).

참고	BASIC 인증 대신 Windows NT 시도 / 응답 (HTTP 응답 코드 401, WWW- 인증서 :NTLM) 으로 보호되는 Microsoft IIS(Internet Information Server) 이 서비스하는 URL 보기는 지원되지 않습니다 .

관리 콘솔에서 [액세스 목록] 서비스를 사용하여 단일 사인온을 사용할 수도 있습니다 . 단일 사인온의 사용에 대한 자세한 내용은 235 페이지의 " 단일 사인온 관리 " 를 참조하십시오 .

➤ HTTP 기본 인증을 사용하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집 …] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [HTTP 기본 인증 사용] 확인란을 선택하여 HTTP 기본 인증의 사용을 설정합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

HTTP 지속 연결 사용

게이트웨이에서 HTTP 지속 연결을 사용하여 웹 페이지의 모든 개체 (이미지 및 스타일 시트 등) 에 대해 소켓이 열리는 것을 방지할 수 있습니다 .

➤ HTTP 지속 연결을 사용하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [HTTP 지속 연결 사용] 확인란을 선택하여 HTTP 연결을 설정합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

지속 연결당 최대 요청 수 지정

➤ **지속 연결당 최대 요청 수를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [지속 연결당 최대 요청 수] 필드로 스크롤하여 필요한 요청 수를 입력합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

지속성 소켓 연결에 시간 초과 지정

➤ **지속성 소켓 연결의 시간 초과를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [지속 연결의 소켓 시간 초과] 필드로 스크롤한 다음 필요한 시간 초과 값을 초 단위로 입력합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

반환 시간을 위한 계정의 유예 시간 초과 지정

유예 시간 초과 반환 시간은 다음의 합입니다 .

- 브라우저가 요청을 보낸 후 이 요청이 게이트웨이에 도달하는 시간 .
- 응답을 보내는 게이트웨이와 이를 실제로 수신하는 브라우저 사이의 시간 .

이 시간은 네트워크 상태 및 클라이언트의 연결 속도와 같은 인자에 의해 결정됩니다 .

➤ **반환 시간을 위한 계정의 유예 시간 초과를 지정하려면**

이것은 클라이언트 (브라우저) 와 게이트웨이 사이에서 네트워크 트래픽의 왕복 시간입니다 .

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [핵심] 탭을 클릭합니다 .

6. [반환 시간을 위한 계정의 유예 시간 초과] 필드로 스크롤하여 필요한 유예 시간 초과를 초 단위로 입력합니다 .

7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

사용자 세션 쿠키를 URL 로 전달 목록 만들기

포털 서버는 사용자 세션을 추적하기 위해 쿠키를 사용합니다 . 이 쿠키는 게이트웨이가 서버에 HTTP 요청을 보낼 때 (예를 들어 , 사용자의 데스크탑 페이지를 생성하기 위해 데스크탑 서블릿이 호출될 때) 서버로 전달됩니다 . 서버의 응용프로그램은 쿠키를 사용하여 사용자를 검증하고 신원을 확인합니다 .

[사용자 세션 쿠키가 전달될 사용자 세션] 목록에 서버 외의 시스템 URL 을 지정하지 않은 경우에는 Portal Server 의 쿠키가 이러한 시스템에 대한 HTTP 요청으로 전달되지 않습니다 . 따라서 이 목록에 URL 을 추가하면 서블릿과 CGI 가 Portal Server 의 쿠키를 받아 API 를 통해 사용자를 식별할 수 있습니다 .

URL 은 뒤에 오는 암시적 와일드카드를 사용하여 매칭됩니다 . 예를 들어 , 목록의 기본 입력

```
http://server:8080
```

은 쿠키가 `http://server:8080` 으로 시작되는 모든 URL 로 전달되도록 합니다 .

다음을 추가하면

```
http://newmachine.eng.siroe.com/subdir
```

쿠키가 이 정확한 문자열로 시작하는 모든 URL 로 전달되도록 합니다 .

이 예에서 , 문자열 "http://newmachine.eng/subdir" 은 전달 목록의 정확한 문자열로 시작하지 않기 때문에 쿠키는 이 문자열로 시작되는 URL 로 전달되지 않습니다 . 쿠키가 이러한 변형된 컴퓨터 이름으로 시작되는 URL 로 전달되도록 하려면 전달 목록에 추가 항목을 추가해야 합니다 .

마찬가지로 , 쿠키는 적합한 항목이 추가되지 않는다면 "https://newmachine.eng.siroe.com/subdir" 로 시작되는 URL 로 전달되지 않습니다 .

➤ 쿠키 URL 전달을 추가하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [사용자 세션 쿠키가 전달될 사용자 세션] 편집 상자로 스크롤하고 필요한 URL 을 입력합니다 .
7. [추가] 를 눌러 이 항목을 [사용자 세션 쿠키가 전달될 사용자 세션] 목록에 추가합니다 .
8. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
9. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

최대 연결 대기 길이 지정

게이트웨이가 허용해야 하는 최대 동시 연결 수를 지정할 수 있습니다 . 이 한계를 벗어난 연결 시도는 게이트웨이에서 허용되지 않습니다 .

➤ 최대 연결 대기 길이를 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [핵심] 탭을 클릭합니다 .

6. [최대 연결 대기 길이] 필드로 스크롤하여 필요한 연결 수를 지정합니다 .

7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

게이트웨이 시간 초과 지정

게이트웨이와 브라우저의 연결이 시간 초과되기까지 걸리는 시간 (밀리초) 을 지정합니다 .

➤ **게이트웨이 시간 초과를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [핵심] 탭을 클릭합니다 .

6. [게이트웨이 시간 초과 (밀리초)] 필드로 스크롤하여 필요한 시간을 밀리초로 지정합니다 .

7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

최대 스레드 풀 크기 지정

게이트웨이 스레드 풀에서 사전에 생성할 수 있는 최대 스레드 수를 지정할 수 있습니다 .

➤ **최대 스레드 풀 크기를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [최대 스레드 풀 크기] 필드로 스크롤하여 필요한 스레드 수를 지정합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

캐시된 소켓 시간 초과 지정

게이트웨이와 Portal Server 와의 연결이 시간 초과되기까지 걸리는 시간 (밀리초) 을 지정합니다 .

➤ **캐시된 소켓 시간 초과를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .

6. [캐시된 소켓 시간 초과] 필드로 스크롤하여 필요한 시간을 밀리초로 지정합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

Portal Server 목록 만들기

서비스 요청에 대한 게이트웨이에 여러 Portal Server 를 구성할 수 있습니다 . 게이트웨이를 설치하는 동안 게이트웨이가 함께 작동해야 하는 Portal Server 를 지정했을 것입니다 . 이 Portal Server 는 기본적으로 Portal Server 필드에 나열됩니다 . http://portal server name:port number 형식으로 목록에 Portal Server 를 더 추가할 수 있습니다 . 게이트웨이는 요청을 처리하기 위해 연속해서 나열된 각 Portal Server 에 접속을 시도합니다 .

➤ 포털 서버를 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집 …] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. Portal Server 필드로 스크롤하여 Portal Server 를 지정합니다 .

   편집 필드에서 http://portal server name:port number 형식으로 Portal Server 를 지정하고 [추가] 를 누릅니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

서버 재시도 간격 지정

이 속성은 Portal Server, Rewriter 프록시 또는 Netlet 프록시를 사용할 수 없게 되는 경우 (충돌이나 다운된 경우 등) 시작 요청 사이의 시간 간격을 지정합니다 .

➤ 포털 서버 재시도 간격을 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [포털 서버 재시도 간격] 필드로 스크롤하여 초 단위 시간을 지정합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

외부 서버 쿠키 저장 활성화

외부 서버 쿠키 저장 옵션을 사용하면 게이트웨이가 게이트웨이를 통해 액세스할 수 있는 타사 응용프로그램이나 서버에 대한 쿠키를 저장하고 관리합니다 . 응용프로그램이나 서버가 쿠키 없는 장치를 서비스할 수 없거나 상태 관리 (레거시 관련 이유로) 를 위해 쿠키에 의존하는 경우에도 게이트웨이는 쿠키 없는 장치를 서비스하고 있다는 사실로부터 응용프로그램이나 서버를 투명하게 숨깁니다 . 쿠키 없는 장치와 클라이언트 검색에 대한 내용은 *Identity Server Customization and API Guide* 를 참조하십시오 .

➤ 외부 서버 쿠키를 저장하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [핵심] 탭을 클릭합니다 .

6. [외부 서버 쿠키 저장] 확인란을 선택하여 외부 서버 쿠키 저장의 사용을 설정합니다 .

7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

URL 에서 세션 얻기

[URL 에서 세션 얻기] 옵션을 선택하면 쿠키 지원 여부에 상관 없이 세션 정보가 URL 의 일부로 인코딩됩니다 . 즉 , 게이트웨이는 클라이언트의 브라우저에서 보내는 세션 쿠키를 사용하지 않고 검증을 위해 URL 에 있는 세션 정보를 사용합니다 .

➤ URL 에서 세션을 얻으려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [핵심] 탭을 클릭합니다 .

6. [URL 에서 세션 얻기] 확인란을 선택하여 URL 에서 세션을 얻습니다 .

7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

쿠키를 안전하다고 표시

쿠키가 안전한 것으로 표시되면 브라우저가 이 쿠키를 추가 보안을 통해 취급합니다. 보안의 구현은 브라우저에 따라 다릅니다 . 이 작업을 위해 [쿠키 관리 사용] 속성을 사용해야 합니다 .

➤ **쿠키를 안전하다고 표시하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [핵심] 탭을 클릭합니다 .
6. [쿠키를 안전하다고 표시] 확인란을 선택하여 쿠키를 안전하다고 표시합니다 .

   [쿠키 관리 사용] 속성이 사용되도록 합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

프록시 탭

게이트웨이 서비스의 [프록시] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- 웹 프록시 사용 활성화
- 웹 프록시에 대한 URL 목록 만들기
- 사용하지 않을 프록시의 URL 목록 만들기
- 도메인 및 부속 도메인의 프록시 목록 만들기
- 프록시 비밀번호 목록 만들기
- 자동 프록시 구성 지원 사용

- 자동 프록시 구성 파일 위치 지정
- 웹 프록시를 통한 Netlet 터널링 활성화

웹 프록시 사용 활성화

➤ **웹 프록시의 사용을 설정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [프록시] 탭을 클릭합니다 .
6. [프록시 사용] 확인란을 선택하여 웹 프록시의 사용을 설정합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

웹 프록시에 대한 URL 목록 만들기

프록시 사용 옵션이 사용 불가능으로 설정되어 있어도 게이트웨이가 [도메인 및 부속 도메인 프록시] 목록에 나열된 웹 프록시를 통해서만 특정 URL 에 접속해야 한다는 것을 지정할 수 있습니다 . [웹 프록시 URL 사용] 필드에서 이러한 URL 을 지정해야 합니다 . 이 값이 프록시 사용에 미치는 영향에 대한 자세한 내용은 56 페이지의 "Identity Server 에 접속하도록 프록시 지정 " 을 참조하십시오 .

➤ **웹 프록시에 대한 URL 을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [프록시] 탭을 클릭합니다 .

6. [웹 프록시 URL 사용] 편집 상자에서 필요한 URL 을 `http://hostname.subdomain.com` 형식으로 입력합니다 . [추가] 를 누릅니다 .

   URL 이 [웹 프록시 URL 사용] 목록에 추가됩니다 .

7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

사용하지 않을 프록시의 URL 목록 만들기

게이트웨이는 [웹 프록시 URL 사용 안함] 목록에 나열된 URL 에 직접 연결을 시도합니다 . 웹 프록시는 이러한 URL 에 연결하는데 사용되지 않습니다 .

➤ **사용하지 않을 URL 을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [프록시] 탭을 클릭합니다 .

6. [웹 프록시 URL 사용 안함] 편집 상자에 필요한 URL 을 입력하고 [추가] 를 클릭합니다 .

   URL 이 [웹 프록시 URL 사용 안함] 목록에 추가됩니다 .

7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

도메인 및 부속 도메인의 프록시 목록 만들기

➤ **도메인 및 부속 도메인의 프록시를 지정하려면**

프록시 정보가 다양한 호스트에 어떻게 적용되는지 자세히 알아보려면 56 페이지의 "Identity Server 에 접속하도록 프록시 지정 " 을 참조하십시오 .

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 오른쪽 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필에 대한 [편집 …] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [프록시] 탭을 클릭합니다 .
6. [도메인 및 부속 도메인의 프록시] 편집 상자로 스크롤하여 필요한 정보를 입력하고 [추가] 를 클릭합니다 . [도메인 및 부속 도메인의 프록시] 목록 상자에 항목이 추가됩니다 .

   프록시 정보를 입력하기 위한 형식은 다음과 같습니다 .

   ```
   domainname proxy1:port1|subdomain1 proxy2:port2|subdomain2
   proxy3:port3|* proxy4:port4
   ```

   * 는 * 이후에 정의된 프록시를 특별히 언급된 경우를 제외하고 모든 도메인과 부속 도메인에 사용해야 한다는 것을 나타냅니다 .

   프록시의 포트를 지정하지 않으면 포트 8080 이 기본적으로 사용됩니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

프록시 비밀번호 목록 만들기

프록시 서버가 일부 또는 모든 사이트에 대한 액세스에 인증을 요구하는 경우 게이트웨이에서 지정된 프록시 서버에 인증을 얻기 위해 필요한 사용자 이름과 비밀번호를 지정해야 합니다 .

➤ **프록시 비밀번호를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [프록시] 탭을 클릭합니다 .
6. [프록시 비밀번호 목록] 필드로 스크롤하여 각 프록시 서버에 대한 정보를 입력하고 [추가] 를 클릭합니다 .

   프록시 정보를 입력하기 위한 형식은 다음과 같습니다 .

   `proxyserver|username|password`

   `proxyserver` 는 도메인 및 부속 도메인의 프록시 목록에 정의된 프록시 서버에 해당합니다 .
7. 인증이 필요한 모든 프록시에 단계 6 을 반복합니다 .
8. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
9. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

자동 프록시 구성 지원 사용

[자동 프록시 구성 사용] 옵션을 선택하면 [도메인 및 부속 도메인의 프록시] 필드에 제공된 정보가 무시됩니다 . 게이트웨이에서는 인트라넷 구성에 프록시 자동 구성 (PAC) 파일만 사용합니다 . PAC 파일에 대한 내용은 62 페이지의 " 자동 프록시 구성 사용 " 을 참조하십시오 .

➤ **자동 프록시 구성 지원을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [프록시] 탭을 클릭합니다 .
6. [자동 프록시 구성 지원] 확인란을 선택하여 PAC 지원을 활성화합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

자동 프록시 구성 파일 위치 지정

➤ **자동 프록시 구성 파일 위치를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [프록시] 탭을 클릭합니다 .
6. [자동 프록시 구성 파일 위치] 필드로 스크롤하여 PAC 파일 이름과 위치를 입력합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

웹 프록시를 통한 Netlet 터널링 활성화

➤ **웹 프록시를 통한 터널 Netlet 을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [프록시] 탭을 클릭합니다 .
6. [웹 프록시를 통한 Netlet 터널링 허용] 확인란을 선택하여 터널링을 활성화합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

보안 탭

게이트웨이 서비스의 [보안] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- 인증되지 않은 URL 목록 만들기
- 인증서 사용 가능 게이트웨이 호스트 목록 만들기
- 40 비트 암호화 연결 허용
- SSL 버전 2.0 사용
- SSL 암호 선택 사용
- SSL 버전 3.0 사용
- Null 암호 활성화
- 신뢰할 수 있는 SSL 도메인 목록 만들기
- 개인 디지털 인증서 (PDC) 인증 구성

인증되지 않은 URL 목록 만들기

일부 URL 에 인증이 필요 없음을 지정할 수 있습니다 . 일반적으로 이미지가 있는 디렉토리와 폴더가 이에 해당합니다 .

➤ **인증되지 않은 URL 경로를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [인증되지 않은 URL] 필드로 스크롤하여 필요한 폴더 경로를 `folder/subfolder` 형식으로 입력합니다 .

   정규화되지 않은 URL(예를 들어 , /images) 은 포털 URL 로 취급됩니다 .

   포털이 아닌 URL 을 추가하려면 URL 을 완전히 정규화하십시오 .
6. [추가] 를 눌러 인증되지 않은 URL 목록에 항목을 추가합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

인증서 사용 가능 게이트웨이 호스트 목록 만들기

➤ **게이트웨이에 인증서 사용 가능 호스트 목록을 추가하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .

   모든 서비스가 왼쪽 창에 표시됩니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 게이트웨이 프로필 페이지가 표시됩니다 .
4. 인증서 기반 인증을 사용할 프로필에 대한 [편집…] 을 클릭합니다 .

5. [보안] 탭을 클릭합니다 .
6. 게이트웨이 이름을 [인증서 사용 가능 게이트웨이 호스트] 에 추가합니다 .

   게이트웨이를 `host1.sesta.com` 형식으로 추가합니다 .
7. [추가] 를 누릅니다 .

40 비트 암호화 연결 허용

40 비트 (취약) SSL(Secure Sockets Layer) 연결을 허용하려는 경우에 이 옵션을 선택하십시오 . 이 옵션을 선택하지 않으면 128 비트 연결만 지원됩니다 .

이 옵션을 선택하지 않으면 사용자의 브라우저가 필요한 연결 유형을 지원하도록 구성되어 있어야 합니다 .

참고 Netscape Navigator 4.7x 를 사용하는 경우에는 다음을 수행합니다 .

- [커뮤니케이터] 메뉴의 [도구] 에서 [보안 정보] 를 선택합니다 .
- 왼쪽 창에서 [네비게이터] 링크를 클릭합니다 .
- [고급 보안 (SSL) 구성] 에서 [SSL v2 구성] 또는 [SSL v3 구성] 을 클릭합니다 .
- 이제 필요한 암호가 사용됩니다 .

➤ 40 비트 암호화 연결을 허용하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [40 비트 암호화 허용] 확인란을 선택하여 40 비트 브라우저 연결을 활성화합니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

SSL 버전 2.0 사용

SSL 버전 2.0 을 사용 또는 사용 해제할 수 있습니다 . SSL 2.0 을 사용 해제하면 구식 SSL 2.0 만 지원하는 브라우저가 SRA 에 인증을 얻을 수 없게 됩니다 . 그러면 보안 수준이 높아집니다 .

➤ **SSL 버전 2.0 을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집 …] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [SSL 버전 2.0 사용] 확인란을 선택하여 버전 2.0 의 사용을 설정합니다 .

   이 필드는 기본적으로 선택됩니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

SSL 암호 선택 사용

SRA 는 많은 표준 암호를 지원합니다 . 사전 구성된 모든 암호의 지원을 선택하거나 필요한 암호만 개별적으로 선택할 수 있습니다 . 각 게이트웨이 인스턴스에 특정 SSL 암호를 선택할 수 있습니다 . 선택한 암호가 클라이언트 사이트에 있으면 SSL 핸드쉐이크가 성공적으로 이루어집니다 .

➤ **개별 암호 선택을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [SSL 암호 선택 사용] 필드로 스크롤하고 옵션을 선택합니다 .

   이 옵션을 통해 SSL2, SSL3 및 TLS 암호 목록에서 필요한 암호를 선택할 수 있습니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

SSL 버전 3.0 사용

SSL 버전 3.0 을 사용 또는 사용 해제할 수 있습니다 . SSL 3.0 을 사용 해제하면 SSL 3.0 만 지원하는 브라우저가 SRA 소프트웨어에 인증을 얻을 수 없게 됩니다 . 그러면 보안 수준이 높아집니다 .

➤ **SSL 버전 3.0 을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [SSL 버전 3.0 사용] 확인란을 선택하여 버전 3.0 의 사용을 설정합니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

Null 암호 활성화

➤ **Null 암호를 활성화하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [Null 암호화 사용] 확인란을 선택하여 Null 암호를 활성화합니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

신뢰할 수 있는 SSL 도메인 목록 만들기

➤ **인증된 SSL 도메인 목록을 만들려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [인증된 SSL 도메인 목록] 으로 스크롤하여 도메인 이름을 입력하고 [추가] 를 클릭합니다 .
6. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

개인 디지털 인증서 (PDC) 인증 구성

PDC 는 인증 기관 (CA) 에서 발행하며 CA 개인 키로 서명됩니다 . CA 는 인증서를 발행하기 전에 요청자의 신원을 검증합니다 . 따라서 PDC 가 있다는 것은 인증 메커니즘이 강력하다는 것을 나타냅니다 .

PDC 에는 소유자의 공용 키 , 소유자 이름 , 만료 날짜 , 디지털 인증서를 발행한 인증 기관의 이름 , 일련 번호와 함께 기타 정보도 포함될 수 있습니다 .

사용자는 PDC 와 스마트 카드 및 Java 카드와 같은 암호화된 장치를 Portal Server 에서 인증을 얻는데 사용할 수 있습니다 . 코드화된 장치는 카드에 저장된 PDC 를 전자적 형태로 보관합니다 . 사용자가 이러한 메커니즘 중 하나를 사용하여 로그인하면 로그인 화면과 인증 화면이 나타나지 않습니다 .

PDC 인증 프로세스에는 다음 단계가 있습니다 .

1. 브라우저에서 사용자가 예를 들어 `https://my.sesta.com`과 같이 연결 요청을 입력합니다 .

   이 요청에 대한 응답은 `my.sesta.com` 에 대한 게이트웨이가 인증서를 허용하도록 구성되었는지 여부에 달려있습니다 .

참고 게이트웨이가 인증서를 허용하도록 구성된 경우 인증서가 있는 로그인만 허용되며 다른 종류의 로그인은 허용되지 않습니다 .

   게이트웨이는 인증서가 알려진 인증 기관에서 발행된 것인지 , 만료되지 않았는지 그리고 위조되지 않았는지 점검합니다 . 인증서가 유효하면 게이트웨이는 사용자를 인증 프로세스의 다음 단계로 진행시킵니다 .

2. 게이트웨이는 인증서를 서버의 PDC 인증 모듈로 전달합니다 .

➤ PDC 및 코드화된 장치를 구성하려면

PDC 및 코드화된 장치의 구성에 다음 단계가 사용됩니다 .

1. Portal Server 시스템의 *portal-server-install-root*`/SUNWam/config/AMConfig-`*instance-name*`.properties` 파일에 다음 행을 추가합니다 .

   ```
   com.iplanet.authentication.modules.cert.gwAuthEnable=yes
   ```

   (파일의 아무 위치에나 추가)

2. PDC 를 사용할 게이트웨이의 인증서 데이터베이스로 필요한 인증서를 가져옵니다 .

   자세한 내용은 7 장 " 인증서 " 를 참조하십시오 .

3. 인증서를 등록합니다 .

   a. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

   b. [Identity 관리] 탭을 선택합니다 .

   c. 조직을 선택합니다 .

   d. [보기] 드롭다운 메뉴에서 [서비스] 를 클릭합니다 .

   e. 핵심 옆의 화살표를 클릭합니다 .

   f. [조직 인증 모듈] 목록 상자 LDAP 에서 인증서와 LDAP 를 선택합니다 .

   g. [사용자 프로필에서 동적으로 생성] 드롭다운 메뉴를 선택합니다 .

   h. [저장] 을 누릅니다 .

4. 신뢰할 수 있는 원격 호스트 목록을 만듭니다 .

   a. [서비스 구성] 탭을 클릭합니다 .

   b. [인증 구성] 옆의 화살표를 누릅니다 .

      아래 인증서 .

   c. [인증된 원격 호스트] 목록 상자로 스크롤합니다 .

   d. ' 없음 ' 을 반전 표시하고 [제거] 를 누릅니다 .

   e. 텍스트 상자에 'any' 를 입력합니다 .

   f. [추가] 를 누릅니다 .

   g. [저장] 을 누릅니다 .

5. 새 인스턴스를 만듭니다 .

   a. [Identity 관리] 탭을 클릭합니다 .

   b. [보기] 드롭다운 메뉴에서 [서비스] 를 선택합니다 .

   c. [인증 구성] 옆의 화살표를 클릭합니다 .

   d. 서비스 인스턴스 목록이 표시됩니다 .

   e. [새로 만들기] 를 누릅니다 .

   f. 새 서비스 인스턴스 페이지가 표시됩니다 .

g. 서비스 인스턴스 이름으로 gatewaypdc 를 입력합니다 .

h. 참고 : 이 이름을 사용해야 합니다 .

i. [제출] 을 클릭합니다 .

j. gatewaypdc 서비스 인스턴스 목록이 표시됩니다 .

k. gatewaypdc 를 눌러 서비스를 편집합니다 .

l. gatewaypdc 가 속성 페이지를 표시합니다 .

m. [인증 구성] 옆의 [편집] 링크를 누릅니다 .

n. 팝업 창이 나타납니다 .

o. [추가] 를 누릅니다 .

구성 *YourOrganization* 에 대한 인증 페이지가 표시됩니다 .

p. [추가] 를 누릅니다 .

q. [인증 모듈 추가] 페이지가 표시됩니다 .

r. [모듈 이름] 필드에서 인증서를 선택하고 [적용 기준] 필드에서 REQUIRED 를 선택합니다 .

s. [확인] 을 누릅니다 .

t. [확인] 을 다시 누르고 팝업 창을 닫습니다 .

6. 인증서를 게이트웨이 호스트에 연결합니다 .

a. [서비스 구성] 탭을 선택합니다 .

b. 게이트웨이 옆의 화살표를 누릅니다 .

오른쪽 창에 게이트웨이 프로필이 표시됩니다 .

c. 게이트웨이 프로필을 선택합니다 .

d. [보안] 탭을 누릅니다 .

e. 게이트웨이 이름을 [인증서 사용 가능 게이트웨이 호스트] 목록 상자에 추가합니다 .

f. [저장] 을 누릅니다 .

g. 서버를 다시 시작합니다 .

h. 단말기 창에서 게이트웨이를 다시 시작합니다 .

gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

7. CA 에서 발행된 클라이언트 인증서를 PDC 사용 가능 게이트웨이에 액세스하는 브라우저에 설치합니다 .
8. 게이트 프로필과 조직에 액세스합니다 .

   https://*gateway:instance-port/YourOrganization*

   인증서 이름으로 사용자 이름과 비밀번호 프롬프트 없이 로그인되어 있어야 합니다 .

Rewriter 탭

게이트웨이 서비스의 [Rewriter] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- 모든 URL 의 다시 쓰기 사용
- 규칙 집합에 대한 URI 의 매핑 목록 만들기
- 구문 분석할 MIME 유형 목록 만들기
- 기본 도메인 및 부속 도메인 지정
- 다시 쓰지 않을 URI 목록 만들기
- MIME 추측 활성화
- 구문 분석할 URI 매핑 목록 만들기
- 마스크 활성화
- 마스킹용 씨드 문자열 지정
- 마스크하지 않을 URI 목록 만들기
- 게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 만들기

모든 URL 의 다시 쓰기 사용

게이트웨이 서비스에서 [모든 URL 다시 쓰기 사용] 옵션을 설정하면 [도메인 및 부속 도메인의 프록시] 목록에 있는 항목을 확인하지 않고 Rewriter 가 모든 URL 을 다시 씁니다 . [도메인 및 부속 도메인의 프록시] 목록에 있는 항목은 무시됩니다 .

➤ **게이트웨이가 모든 URL 을 다시 쓰도록 하려면**

1. Sun Java™ System Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .

3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 대한 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [Rewriter] 탭 아래의 [기본] 부분을 클릭합니다 .

6. [모든 URL 다시 쓰기 사용] 확인란을 선택하여 게이트웨이가 모든 URL 을 다시 쓸 수 있게 합니다 .

7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

규칙 집합에 대한 URI 의 매핑 목록 만들기

규칙 집합은 Identity Server 관리 콘솔의 포털 서버 구성 아래의 Rewriter 서비스에서 만들어집니다 . 자세한 내용은 ***Portal Server** 관리 설명서*를 참조하십시오 .

규칙 집합을 만든 후 [규칙 집합에 URI 매핑] 필드를 사용하여 도메인과 규칙 집합을 연관시킵니다 . 기본적으로 다음 두 항목이 [규칙 집합에 URI 매핑] 필드에 추가됩니다 .

- *://*.Sun.COM/portal/*|default_gateway_ruleset

  여기서 sun.com 은 포털의 설치 도메인이고 /portal 은 포털 설치 컨텍스트입니다 .

- *|generic_ruleset

이것은 기본 도메인의 모든 페이지에 대해 기본 게이트웨이 규칙 집합이 적용된다는 것을 의미합니다 . 기타 모든 페이지에는 일반 규칙 집합이 적용됩니다 . 기본 게이트웨이 규칙 집합과 일반 규칙 집합은 사전 구성된 규칙 집합입니다 .

참고 데스크탑에 나타나는 모든 컨텐트에는 컨텐트가 어디서 가져왔는지 상관 없이 기본 도메인의 규칙 집합이 사용됩니다 .

예를 들어 , 데스크탑이 URL `yahoo.com` 에서 컨텐트를 스크랩하도록 구성되었다고 가정합시다 . Portal Server 는 `sesta.com`에 있습니다 . `sesta.com` 에 대한 규칙 집합이 불려온 컨텐트에 적용됩니다 .

참고 규칙 집합을 지정하는 도메인은 [도메인 및 부속 도메인의 프록시] 목록에 있어야 합니다 .

➤ URI 를 규칙 집합에 매핑하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [기본] 부분을 클릭합니다 .
6. [규칙 집합에 URI 매핑] 필드로 스크롤합니다 .
7. 필요한 도메인이나 호스트 이름 그리고 규칙 집합을 [규칙 집합에 URI 매핑] 필드에 입력하고 [추가] 를 누릅니다 .

   항목이 [규칙 집합에 URI 매핑] 필드에 추가됩니다 .

   도메인이나 호스트 이름 그리고 규칙 집합을 지정하는 형식은 다음과 같습니다 .

   ```
   domain name|ruleset name
   ```

   예 :

   ```
   eng.sesta.com|default
   ```

참고 규칙 집합을 적용하는 우선 순위는 hostname-subdomain-domain 입니다 .

예를 들어 , 도메인 기반 규칙 집합 목록에 다음 항목이 있다고 가정합니다 .

sesta.com|ruleset1

eng.sesta.com|ruleset2

host1.eng.sesta.com|ruleset3

ruleset3 은 host1 의 모든 페이지에 적용됩니다 .

ruleset2 는 host1 에서 가져온 페이지를 제외하고 eng 부속 도메인의 모든 페이지에 적용됩니다 .

ruleset1 은 eng 부속 도메인과 host1 에서 가져온 페이지를 제외하고 sesta.com 도메인의 모든 페이지에 적용됩니다 .

8. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
9. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

Outlook Web Access 의 규칙 집합

SRA 소프트웨어는 MS Exchange 2000 SP3 및 Outlook Web Access (OWA) 의 MS Exchange 2003 을 지원합니다 .

➤ OWA 규칙 집합을 구성하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .

5. [규칙 집합에 URI 매핑] 필드에서 Exchange 2000 이 설치된 서버 이름에 이어 Exchange 2000 서비스 팩 4 OWA 규칙 집합의 이름을 입력합니다 .

   예 :

   exchange.domain.com|exchange_2000sp3_owa_ruleset.

구문 분석할 MIME 유형 목록 만들기

Rewriter 에는 컨텐트 유형 (HTML, JAVASCRIPT, CSS 및 XML) 에 따라 웹 페이지의 구문을 분석하기 위한 4 가지 구문 분석기가 있습니다 . 기본적으로 공통 MIME 유형이 이러한 구문 분석기와 연결되어 있습니다 . 게이트웨이 서비스의 [구문 분석기를 MIME 유형에 매핑] 필드에서 새로운 MIME 유형을 이런 구문 분석기와 연결시킬 수 있습니다 . 그러면 Rewriter 의 기능이 다른 MIME 유형까지 확장됩니다 .

여러 항목을 입력할 때는 세미콜론이나 콤마로 구분합니다 (";" 또는 ",").

예 :

HTML=text/html;text/htm;text/x-component;text/wml; text/vnl/wap.wml

이것은 이러한 MIME 을 가진 모든 컨텐트가 HTML Rewriter 로 보내지고 URL 을 다시 쓰도록 HTML 규칙이 적용된다는 의미입니다 .

팁 MIME 매핑에서 불필요한 구문 분석기를 제거하면 작동 속도를 높일 수 있습니다 . 예를 들어 , 특정 인트라넷의 컨텐트에 JavaScript 가 없다는 것이 확실하면 MIME 매핑 목록에서 JAVASCRIPT 항목을 제거할 수 있습니다 .

➤ **MIME 매핑을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [기본] 부분을 클릭합니다 .

6. [구문 분석기를 MIME 유형에 매핑] 필드로 스크롤하여 편집 상자에 필요한 MIME 유형을 추가합니다 . 여러 항목을 구분할 때는 세미콜론이나 콤마를 사용합니다 .

   항목을 `HTML=text/html;text/htm` 형식으로 지정합니다 .

7. [추가] 를 눌러 목록에 필요한 항목을 추가합니다 .
8. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
9. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

기본 도메인 및 부속 도메인 지정

기본 도메인 및 부속 도메인은 URL 에 도메인과 부속 도메인 없이 호스트 이름만 있을 때 유용합니다 . 이 경우에 게이트웨이는 호스트 이름이 기본 도메인 및 부속 도메인에 있다고 가정하고 그에 따라 진행합니다 .

예를 들어 , URL 의 호스트 이름이 `host1` 이고 기본 도메인과 부속 도메인이 `red.sesta.com` 으로 지정된 경우 , 호스트 이름은 `host1.red.sesta.com` 으로 확인됩니다 .

➤ 기본 도메인 및 부속 도메인을 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 클릭합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 오른쪽 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필 옆에 대한 [편집 …] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .

5. [Rewriter] 탭 아래의 [기본] 부분을 클릭합니다 .
6. [기본 도메인] 필드로 스크롤하여 `subdomain.domain name` 형식으로 필요한 기본값을 입력합니다 .
7. 게이트웨이 프로필 편집 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

다시 쓰지 않을 URI 목록 만들기

➤ **기본 도메인 및 부속 도메인을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [다시 쓰지 않을 URI] 필드로 스크롤하여 편집 상자에 URI 를 추가합니다 .

   참고 : 이 목록에 #* 를 추가하면 href 규칙이 규칙 집합의 일부라 하더라도 URI 를 다시 쓸 수 있습니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

MIME 추측 활성화

Rewriter 는 페이지의 MIME 유형에 따라 구문 분석기를 선택합니다 . WebLogic 및 Oracle 같은 일부 웹 서버는 MIME 유형을 보내지 않습니다 . 이 문제를 해결하려면 [구문 분석기와 URI 매핑] 목록 상자에 데이터를 추가하여 MIME 추측 기능을 활성화할 수 있습니다 .

➤ **MIME 추측을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .

5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [MIME 추측 사용] 확인란을 선택하여 MIME 추측의 사용을 설정합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

구문 분석할 URI 매핑 목록 만들기

[MIME 추측] 확인란이 선택된 상태에서 서버가 MIME 유형을 보내지 않으면 구문 이 상자를 사용하여 분석기를 URI 에 매핑합니다 .

각 URI 는 세미콜론으로 구분합니다 .

예 : HTML=*.html; *.htm;*Servlet

이 것은 html, htm 또는 Servlet 확장을 가진 모든 페이지에 대한 컨텐트를 다시 쓰기 위해 HTML Rewriter 가 사용된다는 것을 의미합니다 .

➤ URI 매핑을 구문 분석하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [구문 분석기를 MIME 유형에 매핑] 필드로 스크롤하여 편집 상자에 데이터를 추가합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

마스크 활성화

마스크를 사용하면 Rewriter 에서 페이지의 인트라넷 URL 이 보이지 않도록 URI 를 다시 쓸 수 있습니다 .

➤ **마스크를 활성화하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [마스크 사용] 확인란을 선택하여 마스크를 활성화합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

마스킹용 씨드 문자열 지정

씨드 문자열은 URI 의 마스크에 사용됩니다 . 씨드 문자열은 마스크 알고리즘에 의해 생성되는 임의 문자열입니다 .

참고 이 씨드 문자열이 변경되거나 게이트웨이가 다시 시작되면 마스크된 URI 를 책갈피에 추가하지 못할 수 있습니다 .

➤ **마스킹용 씨드 문자열을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .

5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [마스킹용 씨드 문자열] 필드로 스크롤하여 편집 상자에 문자열을 추가합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

마스크하지 않을 URI 목록 만들기

일부 응용 프로그램 (애플릿 등) 은 인터넷 URI 가 필요하기 때문에 마스크할 수 없습니다 . 이러한 응용프로그램을 지정하려면 URI 를 목록 상자에 추가합니다 .

예를 들어 다음을 목록 상자에 추가하면

```
*/Applet/Param*
```

컨텐트 URI `http://abc.com/Applet/Param1.html` 이 규칙 집합의 규칙에서 매칭되는 경우 URL 이 마스크되지 않습니다 .

➤ 마스크하지 않을 URI 를 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .

4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .

5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [마스크하지 않을 URI 목록] 필드로 스크롤하여 편집 상자에 URI 를 추가합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 만들기

게이트웨이가 HTTP 와 HTTPS 모드 모두에서 실행되는 경우 Rewriter 가 일관된 프로토콜을 사용하여 HTML 컨텐트의 참조 리소스에 액세스하게 할 수 있습니다 .

예를 들어 원본 URL 이 `http://intranet.com/Public.html` 이라면 http 게이트웨이가 추가됩니다 . 원본 URL 이 `https://intranet.com/Public.html` 이라면 https 게이트웨이가 추가됩니다 .

참고 이는 Javascript 로 생성된 동적 URI 가 아닌 정적 URI 에만 적용됩니다 .

➤ **게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 프로필 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필을 클릭합니다 .

   게이트웨이 - *gateway-profile-name* 페이지가 나타납니다 .
5. [Rewriter] 탭 아래의 [고급] 부분을 클릭합니다 .
6. [게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 표시] 확인란을 선택합니다 .
7. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
8. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

기록 탭

게이트웨이 서비스의 [기록] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- 기록 사용
- Netlet 기록 사용

기록 사용

게이트웨이 기록 파일이 각 세션에 대한 최소 정보 또는 세부 정보를 포착하도록 지정할 수 있습니다. 기록 정보는 Identity 서버 구성 속성의 기록 부분의 일부로 로그 위치 속성에 지정된 디렉토리에 저장됩니다. 이 기록은 Portal Server 컴퓨터에 위치합니다.

기록 이름에는 다음 방식이 사용됩니다.

`srapGateway_`*gatewayhostname_gateway-profile-name*

기록 정보는 Identity 서버 구성에 지정된 대로 파일로 저장하거나 데이터베이스로 저장할 수 있습니다. 기록의 각 필드는 콤마로 구분된 ASCII 값이며 다른 데이터 분석 도구로 내보낼 수 있습니다.

➤ 게이트웨이 기록을 사용하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다.
2. [서비스 구성] 탭을 선택합니다.
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다.

   게이트웨이 페이지가 표시됩니다.
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다.

   게이트웨이 프로필 편집 페이지가 표시됩니다.
5. [기록 사용] 확인란을 선택하여 게이트웨이 기록의 사용을 설정합니다.
6. [세션 기록마다 사용] 확인란을 선택하여 클라이언트 주소, 요청 유형 및 대상 호스트와 같은 최소 기록 정보를 포착합니다.

참고 기록 정보는 [기록 사용] 필드가 사용되고 있는 경우에만 포착됩니다.

7. 게이트웨이가 클라이언트, 요청 유형, 대상 호스트, 요청의 유형, 클라이언트 요청 URL, 클라이언트 사후 데이터 크기, 세션 아이디, 요청 결과 코드 및 전체 응답 크기와 같은 세부적 기록 정보를 포착하도록 [세션 기록마다 상세 정보 표시 사용] 을 선택합니다.

참고 세부 기록 정보는 [세션 기록마다 사용] 확인란이 사용되고 있는 경우에만 포착됩니다.

8. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
9. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

Netlet 기록 사용

이 옵션을 선택하여 Netlet 관련 작업을 기록할 수 있습니다 . Netlet 기록에는 Netlet 세션에 대한 다음 세부 사항이 들어갑니다 .

- 시작 시간
- 소스 주소
- 소스 포트
- 서버 주소
- 서버 포트
- 중단 시간
- 상태 (시작 또는 중단)

➤ **Netlet 기록을 사용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. SRA 구성 아래에서 게이트웨이 옆에 있는 화살표를 클릭합니다 .

   게이트웨이 페이지가 표시됩니다 .
4. 속성을 설정할 게이트웨이 프로필 옆에 있는 [편집…] 을 클릭합니다 .

   게이트웨이 프로필 편집 페이지가 표시됩니다 .
5. [Netlet 기록 사용] 확인란을 선택하여 Netlet 기록의 사용을 설정합니다 .
6. 페이지 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .
7. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

NetFile 구성

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 NetFile 을 구성하는 방법에 대해 설명합니다 .

참고 Identity Server 관리 콘솔의 맨 위 오른쪽 구석에서 [도움말] 을 누르고 [SRA 도움말] 을 누르면 모든 SRA 속성을 빠르게 참조할 수 있습니다 .

➤ **NetFile 속성을 구성하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
8. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

   여기서 해당 탭을 클릭합니다 .

   - 호스트 탭
   - 권한 탭
   - 보기 탭
   - 작업 탭

- 일반 탭

각각에서 구성할 수 있는 탭과 속성을 아래에서 설명합니다 .

호스트 탭

NetFile 서비스의 [호스트] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- OS 문자 집합 지정
- 호스트 검색 순서 지정
- 공통 호스트 목록 구성
- 기본 도메인 지정
- Windows 도메인 / 워크그룹 지정
- 기본 WINS/DNS 서버 지정
- 다른 유형의 호스트에 액세스 지정
- 허용된 호스트 목록 구성
- 거부된 호스트 목록 구성

OS 문자 집합 지정

호스트와의 통신을 위한 기본 인코딩으로 사용할 문자 집합을 지정할 수 있습니다 . 기본값은 UTF-8 입니다 . 참조

주의	문자 집합을 올바로 지정하지 않으면 컴퓨터의 작동과 나타나는 오류 메시지를 예측할 수 없습니다 .

➤ OS 문자 집합을 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .

5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [호스트] 탭을 누르면 [구성] 이 있습니다 .
12. [OS 문자 집합] 필드로 스크롤하여 문자 집합 코드를 선택합니다 .
13. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

호스트 검색 순서 지정

➤ **호스트 검색 순서를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .

11. [호스트] 탭을 누르면 [구성] 이 있습니다 .

12. [호스트 검색 순서] 필드로 스크롤하여 호스트 유형을 선택합니다 .

13. 위쪽 및 아래쪽 버튼을 사용하여 호스트 검색 순서를 변경합니다 .

14. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

공통 호스트 목록 구성

모든 원격 NetFile 사용자가 NetFile 을 통해 이용할 수 있는 호스트 목록을 구성할 수 있습니다 . 추가하는 각 호스트에 다음 정보를 지정해야 합니다 .

호스트 이름 - 간단한 호스트 이름 또는 정규 이름을 입력할 수 있습니다 . 제공한 호스트 이름이 사용자가 구성한 호스트 이름과 일치하면 두 정보 집합이 병합되고 사용자 지정 값이 우선적으로 적용됩니다 .

예를 들어 `sesta`, `siroe`, `florizon` 및 `abc` 의 4 가지 공통 호스트를 구성했다고 가정합시다 . 사용자가 3 개의 호스트를 구성하고 이중 2 개가 `sesta` 와 `siroe` 입니다 . 이러한 충돌 상황에서 사용자가 지정한 값이 관리자가 지정한 값보다 우선합니다 . `florizon` 과 `abc` 는 사용자의 NetFile 에도 나열되며 사용자는 이 호스트에서 다양한 작업을 수행할 수 있습니다 . `florizon` 을 거부된 호스트 목록에 포함시킨 경우는 `florizon` 이 사용자의 NetFile 에 나열은 되지만 `florizon` 에서 작업을 수행할 수 없습니다 .

호스트 유형 - 사용자가 공통 호스트 목록에 나열된 호스트를 이미 추가했다면 사용자 설정이 우선합니다 . 유형이 충돌하면 해당 사용자에 대해 관리자가 추가한 공유가 추가되지 않습니다 . 사용자와 관리자가 같은 공유를 추가하면 공유가 추가되지만 사용자가 설정한 비밀번호가 우선합니다 .

인코딩 - 여기서 지정한 값과 사용자 설정 사이에 충돌이 있으면 사용자 설정이 우선합니다 . 비었거나 잘못된 설정을 지정하면 클라이언트 OS(사용자의 시스템) 의 문자 집합을 사용합니다 .

참고 사용자는 NetFile 클라이언트 응용프로그램에서 이러한 값을 편집할 수 있습니다 . 그러나 편집된 값은 현재 세션에만 적용됩니다 . 사용자가 로그아웃한 다음 다시 로그인하면 편집된 값이 유지되지 않습니다 .

➤ 공통 호스트 목록을 구성하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [호스트] 탭을 누르면 [구성] 이 있습니다 .

    NetFile > AddNetFile 호스트 페이지가 표시됩니다 .
12. 공통 호스트를 추가하려면

    a. 다음 필드에 필요한 정보를 입력합니다 .

    - 호스트 이름
    - 호스트 종류
    - 인코딩
    - Windows 도메인 / 워크그룹
    - 사용자 이름
    - 비밀번호

    b. 추가할 각 공유에 대해 다음 필드에 필요한 정보를 입력하고 [목록에 추가] 를 클릭합니다 .

    - 공유 목록
    - 공유 이름

- 공유 비밀번호

c. [확인] 을 누릅니다 .

d. 추가하거나 삭제할 각 공통 호스트에 대해 이 과정을 반복합니다 .

13. 공통 호스트 목록에서 공통 호스트를 삭제하려면

a. [삭제] 를 누르고 공유 목록에서 호스트 이름을 선택합니다 . 그런 다음 [제거] 를 클릭합니다 .

b. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

기본 도메인 지정

허용된 호스트에 접속하기 위해 NetFile 이 사용해야 하는 기본 도메인을 지정할 수 있습니다 .

이 기본 도메인 값은 사용자가 NetFile 을 사용하여 호스트를 추가하면서 정규 호스트 이름을 지정하지 않은 경우에만 적용할 수 있습니다 .

주의 [기본 도메인] 필드가 비어있지 않고 유효한 도메인 이름이 들어 있는지 확인합니다 .

➤ **기본 도메인을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [Identity 관리] 탭을 선택합니다 .

3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .

4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .

5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .

6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .

7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .

8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .

9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [호스트] 탭을 누르면 [구성] 이 있습니다 .
12. [기본 도메인] 필드로 스크롤하여 기본 도메인 이름을 입력합니다 .
13. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

Windows 도메인 / 워크그룹 지정

이것은 사용자가 Windows 호스트에 액세스할 때 선택하는 기본 Windows 도메인 또는 워크그룹입니다 .

사용자가 컴퓨터를 추가하면서 다른 값을 지정하여 이 값을 무시할 수 있습니다 .

➤ **기본 Windows 도메인 또는 워크그룹을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [호스트] 탭을 누르면 [구성] 이 있습니다 .
12. [기본 Windows 도메인 / 워크그룹] 필드로 스크롤하여 기본 도메인 또는 워크그룹 이름을 입력합니다 .

13. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

기본 WINS/DNS 서버 지정

이것은 NetFile 이 Windows 호스트에 액세스할 때 사용하는 WINS/DNS 서버입니다 .

➤ **기본 WINS/DNS 서버를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [호스트] 탭을 누르면 [구성] 이 있습니다 .
12. [기본 WINS/DNS 서버] 필드로 스크롤하여 기본 Windows 또는 DNS 서버 이름을 입력합니다 .
13. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

다른 유형의 호스트에 액세스 지정

사용자가 Windows, FTP, NFS 또는 Netware 호스트와 같은 특정 호스트에 액세스할 수 있는지 여부를 지정할 수 있습니다 . 각 유형의 호스트에 대한 액세스를 허용하거나 거부하기 위한 옵션을 설정할 수 있습니다 . 기본적으로 이 옵션의 사용이 모두 설정됩니다 .

➤ **다른 유형의 호스트에 액세스를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 . [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
3. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
4. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
5. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
6. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
7. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

   NetFile 페이지가 표시됩니다 .
8. [호스트] 탭 , [액세스] 하위 부분을 클릭합니다 .
9. 액세스를 활성화할 호스트 유형을 클릭합니다 . 다음을 활성화시킬 수 있습니다 .
   - Windows 호스트에 액세스 허용
   - FTP 호스트에 액세스 허용
   - NFS 호스트에 액세스 허용
   - Netware 호스트에 액세스 허용

   옵션을 선택하면 사용자가 그 특정 호스트 유형에 액세스할 수 있게 됩니다 . 확인란을 지우면 사용자가 그 호스트 유형에 액세스하지 못합니다 .
10. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

허용된 호스트 목록 구성

이 목록에 * 항목이 있기 때문에 사용자는 기본적으로 NetFile 을 통해 모든 호스트에 액세스할 수 있습니다 . 이를 변경하려면 * 항목을 제거하고 사용자가 NetFile 을 통해 액세스해야 하는 호스트만 이 목록에서 지정합니다 . 또는 , * 입력을 그대로 두고 [거부된 호스트] 목록에서 액세스를 거부할 호스트를 지정할 수 있습니다 . 이 경우 [거부된 호스트] 목록에 지정된 호스트를 제외한 모든 호스트에 액세스가 허용됩니다 .

자세한 내용은 295 페이지의 " 거부된 호스트 목록 구성 " 를 참조하십시오 .

참고 [허용된 호스트] 및 [거부된 호스트] 목록이 모두 비어 있으면 어떤 호스트에도 액세스가 허용되지 않습니다 .

➤ 허용된 호스트 목록을 만들려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [호스트] 탭 , [액세스] 하위 부분을 클릭합니다 .
12. [허용된 호스트] 필드로 스크롤합니다 . 편집 필드에서 액세스를 허용할 호스트 이름을 입력하고 [추가] 를 클릭합니다 .

    호스트 이름이 [허용된 호스트] 목록 상자에 추가됩니다 .

13. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

거부된 호스트 목록 구성

288 페이지의 " 공통 호스트 목록 구성 " 에서 공통적으로 사용할 수 있는 호스트 목록을 지정하고 나면 NetFile 을 통한 사용자의 액세스를 거부할 호스트 목록도 지정할 수 있습니다 .

참고 사용자가 NetFile 창에서 이미 추가한 호스트의 액세스를 거부하는 경우에도 거부된 호스트는 사용자의 NetFile 창에 계속 표시됩니다 . 그러나 사용자는 이 호스트에서 어떤 작업도 수행할 수 없습니다 .

NetFile Java2 에서 응용프로그램에 나타나는 경우 거부된 호스트에는 빨간색 십자 모양이 표시되어 액세스할 수 없음을 나타냅니다 .

참고 [허용된 호스트] 및 [거부된 호스트] 목록이 모두 비어 있으면 어떤 호스트에도 액세스가 허용되지 않습니다 .

➤ **거부된 호스트 목록을 만들려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .

10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .

11. [호스트] 탭 , [액세스] 하위 부분을 클릭합니다 .

12. [거부된 호스트] 필드로 스크롤합니다 . 편집 필드에서 액세스를 거부할 호스트 이름을 입력합니다 .

13. [추가] 를 누릅니다 .

    호스트 이름이 [거부된 호스트] 목록 상자에 추가됩니다 .

14. 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

권한 탭

NetFile 서비스의 [권한] 탭을 사용하여 사용자가 원격 호스트에서 다음 작업을 수행할 수 있는 권한을 허용하거나 거부할 수 있습니다 .

- 파일 이름 변경
- 파일 및 폴더 삭제
- 파일 업로드
- 파일 및 폴더 다운로드
- 파일 검색
- 메일로 파일 보내기
- 파일 압축
- 사용자 아이디 변경

  이 환경 설정으로는 사용자가 다른 아이디로 NetFile 을 사용하는 호스트에 연결할 수 있는지 여부를 지정합니다 . 대규모 조직에서 사용자는 여러 개의 사용자 아이디를 가질 수 있습니다 . 사용자가 하나의 사용자 아이디만 사용하도록 제한해야 할 수 있습니다 . 이 경우에 [사용자 아이디 변경 허용] 옵션을 사용 해제할 수 있습니다 . 그러면 특정 조직의 모든 사용자가 해당 사용자 아이디를 변경할 수 없고 하나의 아이디 (데스크탑 로그인 아이디) 로만 NetFile 을 사용하여 호스트에 연결할 수 있습니다 . 또 다른 경우에 , 사용자가 여러 컴퓨터에서 서로 다른 로그인 아이디를 가질 수 있으며 이 때는 사용자가 필요에 따라 아이디를 변경하도록 허용해야 할 수 있습니다 .

- Windows 도메인 변경

  이 옵션은 NT 도메인에 적용할 수 있습니다 .

  사용자가 시스템을 추가하면서 사용자 NT 도메인 이름 필드에 잘못된 도메인 이름을 지정하면 오류 메시지가 표시됩니다 . 사용자가 나중에 호스트 정보를 편집하고 잘못된 도메인 이름을 지정할 경우에는 오류 메시지가 나타나지 않습니다 .

  사용자가 도메인 이름을 지정하면 이 도메인의 아이디와 비밀번호도 지정해야 합니다 . 호스트의 아이디와 비밀번호를 사용해야 하는 경우 사용자가 사용자 NT 도메인 이름 필드에서 도메인을 제거해야 합니다 .

이 권한 옵션은 기본적으로 활성화됩니다 .

➤ **권한을 사용 / 사용 해제하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [권한] 탭을 클릭합니다 .
12. 필요한 [허용] 필드로 스크롤하여 권한 허용을 위한 확인란을 클릭합니다 .
13. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

참고 사용자가 NetFile 을 사용하기 시작한 후에 이러한 옵션을 사용 해제하면 사용자가 NetFile 을 로그아웃하고 다시 로그인하는 경우에만 변경 사항이 적용됩니다 .

보기 탭

NetFile 서비스의 [보기] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- NetFile 창 크기 지정
- NetFile 창 위치 지정

NetFile 창 크기 지정

사용자 데스크탑에서 픽셀 단위로 NetFile 창의 크기를 지정할 수 있습니다 . 기본값은 `700|400` 픽셀입니다 . 잘못된 값을 입력하면 NetFile 이 기본값을 사용합니다 .

참고 사용자도 사용자가 사용할 수 있는 제한된 관리 콘솔에서 이 값을 편집할 수 있습니다 . 사용자가 데스크탑에서 NetFile 창의 크기를 조정하면 지정하는 값이 새 값으로 교체됩니다 .

➤ **NetFile 창의 크기를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
8. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

   NetFile 페이지가 표시됩니다 .
9. [보기] 탭을 클릭합니다 .
10. [창 크기] 필드로 스크롤하여 필요한 창 크기를 픽셀로 입력합니다 .

    공백 없이 `700|400` 형식으로 값을 입력합니다 . 좌표는 `x|y` 형태입니다 . 구분 기호로 다른 문자를 사용해서는 안됩니다 .

11. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

NetFile 창 위치 지정

사용자 데스크탑에서 NetFile 창의 표시 위치를 지정할 수 있습니다 . 기본값은 `100|50` 픽셀입니다 . 잘못된 값을 입력하면 NetFile 이 기본값을 사용합니다 .

참고 사용자도 사용자가 사용할 수 있는 제한된 관리 콘솔에서 이 값을 편집할 수 있습니다 . 사용자가 데스크탑에서 NetFile 창의 위치를 바꾸면 지정하는 값이 새 값으로 교체됩니다 .

➤ **NetFile 창의 위치를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [보기] 탭을 클릭합니다 .
12. [창 위치] 필드로 스크롤하여 필요한 창 위치 좌표를 입력합니다 .

    공백 없이 `100|50` 형식으로 값을 입력합니다 . 좌표는 `x|y` 형태입니다 . 구분 기호로 다른 문자를 사용해서는 안됩니다 .
13. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

작업 탭

NetFile 서비스의 [작업] 탭을 사용하여 다음 작업을 수행할 수 있습니다 .

- 임시 파일 디렉토리 지정
- 파일 업로드 제한 크기 설정
- 디렉토리 검색 제한 지정
- 압축 지정

임시 파일 디렉토리 지정

NetFile 에서 파일 메일링 등의 일부 파일 작업을 수행하려면 임시 디렉토리가 필요합니다 . 기본 임시 디렉토리는 `/tmp` 입니다 . 임시 디렉토리는 필요한 작업이 수행된 다음에 삭제됩니다 .

지정된 임시 디렉토리는 서버에 없는 경우에 만들어집니다 .

웹 서버 실행에 사용하고 있는 아이디 (`nobody` 또는 `noaccess` 등) 에 지정 디렉토리에 대한 `rwx` 권한이 있는지 확인하십시오 . 이 아이디에 필요한 임시 디렉토리의 전체 경로에 대한 `rx` 권한이 있는지도 확인하십시오 .

팁	NetFile 에 별도 임시 디렉토리를 만들어야 하는 경우가 있습니다 . Portal Server 의 모든 모듈에 공통된 임시 디렉토리를 지정하면 디스크 공간이 금방 부족해질 수 있습니다 . 파일 메일링과 같은 NetFile 의 일부 작업은 임시 디렉토리에 공간이 없으면 작동하지 않습니다 .

➤ 임시 디렉토리를 지정하려면

1. Sun Java™ System Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .

7. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
8. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

   NetFile 페이지가 표시됩니다 .
9. [작업] 탭 , [트래픽] 하위 부분을 클릭합니다 .
10. [임시 디렉토리 위치] 필드로 스크롤하여 필요한 임시 디렉토리 위치를 입력합니다 .
11. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

파일 업로드 제한 크기 설정

이 필드에 업로드할 수 있는 최대 파일 크기를 지정할 수 있습니다 . 업로드하는 파일의 크기가 여기에 지정된 값을 초과하면 오류 메시지가 표시되고 파일이 업로드되지 않습니다 . 기본값은 5MB 입니다 . 잘못된 값을 입력하면 NetFile 이 잘못된 값을 기본값으로 재설정합니다 .

사용자마다 다른 파일 업로드 제한 크기를 지정할 수 있습니다 .

참고 업로드의 최대 파일 크기를 MB 로 지정합니다 . 정수 값만 입력해야 합니다 .

➤ **파일 업로드 제한 크기를 설정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
8. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

   NetFile 페이지가 표시됩니다 .

9. [작업] 탭 , [트래픽] 하위 부분을 클릭합니다 .
10. [파일 업로드 제한 크기 (MB)] 필드로 스크롤합니다 . 필요한 제한 크기를 MB 로 입력합니다 .
11. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

디렉토리 검색 제한 지정

한번의 검색으로 검색되는 최대 디렉토리 수를 구성할 수 있습니다 . 이 제한은 많은 사용자가 동시에 로그인하여 네트워크 체증을 유발하고 액세스 속도를 저하시키는 것을 방지합니다 . 기본값은 `100` 입니다 . 잘못된 값을 입력하면 NetFile 이 잘못된 값을 기본값으로 재설정합니다 .

사용자가 A 라는 디렉토리를 가지고 있고 A 에 100 개의 하위 디렉토리가 있다고 가정합시다 . 최대 검색 디렉토리를 100 개로 지정하면 검색 과정이 디렉토리 A 에서 끝납니다 . 디렉토리 A 에서 100 개 제한에 도달하고 나면 사용자 시스템의 다른 디렉토리를 통해 검색이 진행되지 않습니다 . 검색 제한에 도달할 때까지 누적된 검색 결과는 검색이 제한을 초과했음을 알리는 오류 메시지를 사용자에게 표시합니다 . 검색을 계속하려면 사용자가 다음 디렉토리에서 수동으로 검색을 다시 시작해야 합니다 .

검색 작업은 하위 디렉토리 우선 방식으로 수행됩니다 . 즉 , 검색 작업은 사용자가 선택한 디렉토리의 모든 하위 디렉토리를 거친 후에 다음 디렉토리로 이동합니다 .

➤ **디렉토리 검색 제한을 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
8. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .

9. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
10. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
11. [작업] 탭 아래의 [검색] 부분을 클릭합니다 .
12. [검색 디렉토리 제한] 필드로 스크롤하여 필요한 숫자를 입력합니다 .

참고 이 필드에 정수 값만 입력해야 합니다 .

13. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

압축 지정

이 압축 속성은 NetFile Java2 에만 적용됩니다 .

➤ 기본 압축 유형을 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

    NetFile 페이지가 표시됩니다 .
7. [작업] 탭 아래의 [압축] 부분을 클릭합니다 .
8. [기본 압축 유형] 필드로 스크롤합니다 .

    Zip 또는 Gzip 을 선택합니다 .
9. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

일반 탭

NetFile 서비스의 [일반] 탭을 사용하여 MIME 유형 구성 파일 위치를 지정할 수 있습니다 .

MIME 유형 구성 파일 위치 지정

이 정보는 클라이언트 브라우저로 보낼 응답 컨텐트 유형을 결정할 때 필요합니다 . 브라우저는 NetFile 열기 또는 다운로드 작업 중에 파일을 연결시켜야 하는 응용프로그램을 확인하기 위해 이 정보가 필요합니다 . 이 정보는 설치 중에 구성됩니다 .

Portal Server 에 있는 웹 서버의 MIME 유형 파일을 사용해야 하는 경우에는 다음과 같이 위치를 지정합니다 .

portal-server-install-root/SUNWam/servers/*instance-name-of-web-server-machine*/config

참고 MIME 유형 구성 파일 위치 속성은 조직 수준에서만 설정할 수 있습니다 .

➤ MIME 유형 구성 파일의 위치를 지정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
6. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
7. [보기] 목록 상자에서 [서비스] 를 선택합니다 .
8. SRA 구성 아래에서 NetFile 옆에 있는 화살표를 클릭합니다 .

   NetFile 페이지가 표시됩니다 .
9. [일반] 탭을 누릅니다 .

10. [MIME 유형 구성 파일 위치] 필드로 스크롤하여 MIME 유형 구성 파일이 있는 전체 경로를 입력합니다 .
11. NetFile 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

NetFile 에 디버깅 활성화

디버그 정보의 위치는 Portal Server 노드의 `AMConfig-`*`instance-name`*`.properties` 파일에 있는 `com.iplanet.services.debug.directory` 속성 설정에 따라 결정됩니다 .

예를 들어 , `com.iplanet.services.debug.directory` 속성의 값이 다음과 같으면

`/var/opt/SUNWam/debug/`

NetFile 에 대한 디버그 정보는 `/var/opt/SUNWam/debug` 디렉토리에 있는 `srapNetFile` 파일에서 찾을 수 있습니다 .

자세한 내용은 ***Identity Server** 관리 설명서*를 참조하십시오 .

Netlet 구성

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 Netlet 속성을 구성하는 방법에 대해 설명합니다 .

참고 Identity Server 관리 콘솔의 맨 위 오른쪽 구석에서 [도움말] 을 누르고 [SRA 도움말] 을 누르면 모든 SRA 속성을 빠르게 참조할 수 있습니다 .

조직 수준에서 구성할 수 있는 모든 속성은 사용자 수준에서도 구성할 수 있습니다 . 조직, 역할 및 사용자 수준 속성에 대한 자세한 내용은 *Identity Server 관리 설명서*를 참조하십시오 .

Netlet 속성을 구성하려면 다음 단계에 따라 조직 수준에서 속성을 구성합니다 .

1. Sun Java™ System Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   여기서부터 다음 작업을 수행할 수 있습니다 .

   - Netlet 규칙 추가
   - 사용자에 Netlet 서비스 지정
   - Netlet 규칙 추가

- 기존 Netlet 규칙 수정
- Netlet 규칙 삭제

사용자 프로필을 구성하고 Netlet 규칙을 만드는 경우가 아니면 사이트의 요구 사항에 따라 다음 속성을 구성해야 합니다 . 이러한 속성은 조직 또는 사용자 수준에서 구성할 수 있습니다 .

- 기본 암호 지정
- 기본 루프백 포트 할당
- 연결에 대한 재인증 활성화
- 연결에 대한 경고 팝업 대화 상자 활성화
- 포트 경고 대화 상자에 확인란 표시 활성화
- 연결 유지 시간
- 포털 로그아웃할 때 Netlet 종료 옵션 설정
- Netlet 규칙에 대한 액세스 정의
- Netlet 규칙에 액세스 거부
- 호스트에 대한 액세스 허용호스트에 대한 액세스 거부

사용자에 Netlet 서비스 지정

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 .

   선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. 선택된 조직에 대한 [보기] 드롭다운 목록에서 [사용자] 를 선택합니다 .
6. 왼쪽 창에서 필요한 사용자 옆에 있는 화살표를 클릭합니다 .
7. 이 사용자에 대해 아직 Netlet 서비스를 사용할 수 없는 경우 , 이 사용자에 대한 [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
8. [추가] 를 누릅니다 .
9. [사용 가능한 서비스] 목록에서 [Netlet] 을 선택합니다 .

10. [저장] 을 누릅니다 .
11. 이 사용자의 [보기] 드롭다운 목록에서 Netlet 서비스를 선택하면 Netlet 속성을 수정할 수 있습니다 .

Netlet 규칙 추가

Identity Server 관리 콘솔의 [Identity 관리] 탭에서 Netlet 규칙을 전역 수준으로 추가하거나 만들 수 있습니다 . 이러한 규칙은 생성되는 새로운 조직에 상속됩니다 .

조직 , 역할 또는 사용자 수준에서 새 규칙을 만들거나 기존 규칙을 수정할 수도 있습니다 .

➤ **Netlet 규칙을 추가하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. 규칙을 만들려는 대상 조직을 선택합니다 .
4. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
5. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
6. [Netlet 규칙] 필드에서 [추가] 를 클릭합니다 .

   [Netlet 규칙 추가] 페이지가 표시됩니다 . 규칙의 모든 필드에는 필요에 따라 변경할 수 있는 예제 값이 들어가 있습니다 .
7. [규칙 이름] 필드에 규칙에 대한 고유한 이름을 입력합니다 .
8. 필요한 암호를 지정합니다 . 기본 암호를 유지하려면 [기본값] 을 선택합니다 . 사용 가능한 암호 목록에서 선택하려면 [기타] 를 선택합니다 .

   기본 암호에 대한 자세한 내용은 312 페이지의 " 기본 암호를 지정하려면 " 을 참조하십시오 .
9. URL 필드에 불러올 응용프로그램에 대한 URL 을 입력합니다 .
10. 애플릿을 다운로드해야 하는 경우에는 [애플릿 다운로드] 확인란을 선택합니다 . 연관된 편집 상자에서 *local-port:server-host:server-port* 형식으로 애플릿 세부 사항을 입력합니다 .

참고 각 규칙에 고유한 `local port` 를 지정합니다 .

Portal Server 호스트 이외 호스트에서 애플릿을 다운로드해야 하는 경우에만 애플릿 세부사항을 지정해야 합니다 . 확인란을 선택하지 않으면 편집 상자가 비활성화됩니다 . 자세한 내용은 187 페이지의 " 원격 호스트에서 애플릿 다운로드 " 를 참조하십시오 .

11. [세션 연장] 확인란을 선택하여 이 규칙에 해당하는 Netlet 세션이 실행되는 동안에 Portal Server 세션 시간이 연장되도록 합니다 .
12. [로컬 포트] 필드에 Netlet 이 수신할 로컬 포트를 입력합니다 .

    FTP 규칙의 경우 로컬 포트 값이 30021 이어야 합니다 .
13. [대상 호스트] 필드에 항목을 입력합니다 .

    정적 규칙의 경우 , Netlet 연결에 대한 대상 컴퓨터의 호스트 이름을 입력합니다 .

    동적 규칙의 경우 , "TARGET" 을 입력합니다 .
14. [대상 포트] 필드에 대상 호스트의 포트를 입력합니다 .
15. [목록에 추가] 를 눌러 [로컬 포트 대 대상 포트] 필드에 마지막 3 개 항목을 반영합니다 .
16. [저장] 을 누릅니다 .

    규칙이 저장되고 Netlet 페이지로 돌아옵니다 . [Netlet 규칙] 목록에 새 규칙 이름이 표시됩니다 .

기존 Netlet 규칙 수정

관리 콘솔의 [Identity 관리] 탭에서 조직 , 역할 또는 사용자 수준으로 기존 규칙을 수정할 수 있습니다 . 이러한 규칙은 생성되는 새로운 조직에 상속됩니다 .

➤ Netlet 규칙을 수정하려면

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. 규칙을 수정하려는 대상 조직을 선택합니다 .
4. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .

5. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .

6. 수정할 규칙 이름을 클릭합니다 .

   [Netlet 규칙 편집] 페이지가 표시됩니다 .

7. 필요에 따라 변경하고 [저장] 을 클릭합니다 .

   수정된 규칙이 저장되고 Netlet 페이지로 돌아옵니다 .

Netlet 규칙 삭제

관리 콘솔의 [Identity 관리] 탭에서 Netlet 규칙을 전역 수준으로 삭제할 수 있습니다 .

➤ **Netlet 규칙을 삭제하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. 규칙을 삭제하려는 대상 조직을 선택합니다 .
4. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .

5. [Netlet 규칙] 목록에서 삭제할 규칙 옆에 있는 확인란을 선택합니다 .
6. [삭제] 를 누릅니다 .

   선택한 규칙이 [Netlet 규칙] 목록에서 제거됩니다 .

참고 이 부분에서는 조직 수준에서 모든 속성의 구성에 대해 설명합니다 .

기본 암호 지정

Netlet 규칙에 대한 기본 암호를 지정해야 합니다 . 규칙의 일부로 암호를 포함하지 않은 기존 규칙을 사용할 때 유용합니다 . 이 필드는 필수입니다 . 193 페이지의 " 이전 버전과의 호환성 " 을 참조하십시오 .

➤ **기본 암호를 지정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [기본 원시 VM 암호] 또는 [기본 Java 플러그인 암호] 필드로 스크롤하고 드롭다운 목록에서 필요한 암호를 선택합니다 . 지원되는 암호 목록에 대해서는 193 페이지의 " 지원되는 비밀번호 " 를 참조하십시오 .
8. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

기본 루프백 포트 할당

이 속성은 Netlet 을 통해 애플릿을 다운로드할 때 로컬 시스템에서 사용할 포트를 지정합니다 . Netlet 규칙에서 우선하지 않는다면 기본값 58000 이 사용됩니다 .

➤ **기본 루프백 포트를 할당하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [기본 루프백 포트] 필드로 스크롤하고 필요한 포트 번호를 입력합니다 .
8. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

연결에 대한 재인증 활성화

Netlet 연결을 구성해야 할 때마다 사용자가 Netlet 비밀번호를 입력하도록 하려면 이 옵션을 활성화합니다 . 이 옵션을 활성화하면 사용자의 데스크탑에서 연결에 대한 경고 팝업이 표시되지 않습니다 . 자세한 내용은 313 페이지의 " 연결에 대한 경고 팝업 대화 상자 활성화 " 를 참조하십시오 .

이 옵션을 활성화하면 사용자가 Netlet 채널 편집 옵션을 사용하여 재인증 비밀번호를 변경할 수 있습니다 . 초기 비밀번호는 기본적으로 `srap-Netlet` 입니다 .

➤ **연결에 대한 재인증 활성화하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [연결에 대한 재인증] 필드로 스크롤하고 옵션을 선택합니다 .
8. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

연결에 대한 경고 팝업 대화 상자 활성화

이 속성은 사용자가 Netlet 을 사용하여 응용 프로그램을 실행하고 있는데 누군가 수신 포트를 통해 Netlet 에 연결하려 하면 사용자의 데스크탑에 경고 팝업 대화 상자를 표시합니다 . 사용자의 데스크탑에 팝업이 나타나지 않도록 하려면 이 속성의 선택을 해제합니다 .

➤ **연결에 대한 경고 팝업을 비활성화하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .

4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 " 위치 " 로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [연결에 대한 경고 팝업 표시] 확인란을 선택하여 경고 팝업을 활성화합니다 .
8. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

포트 경고 대화 상자에 확인란 표시 활성화

이 속성은 관리 콘솔에서 활성화되어 있는 경우 Netlet 에서 로컬 시스템의 포트에서 자유롭게 이용할 수 있는 포트를 통해 대상 호스트에 연결하려 하면 사용자 데스크탑의 경고 팝업에 확인란을 표시합니다 . 사용자가 데스크탑에서 이 확인란을 선택하거나 선택을 취소하면 그에 따라 팝업을 활성화 또는 비활성화할 수 있습니다 .

관리 콘솔에서 [포트 경고 대화 상자에 확인란 표시] 옵션을 비활성화하면 사용자가 이 경고 팝업이 나타나지 않도록 억제할 수 있습니다 .

➤ **사용자가 포트 경고 대화 상자를 나타나지 않도록 허용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [포트 경고 대화 상자에 확인란 표시] 필드로 스크롤하여 상자 선택을 해제합니다 .
8. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

연결 유지 시간

클라이언트에서 웹 프록시를 통해 게이트웨이에 연결하는 경우에는 유휴 Netlet 연결이 프록시 시간 초과로 해제됩니다 . 이를 방지하려면 이 매개 변수에 프록시 시간 초과보다 작은 값을 지정합니다 .

➤ **연결 유지 시간을 설정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [연결 유지 시간 (분)] 필드로 스크롤하고 필요한 시간 길이를 입력합니다 .
8. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

포털 로그아웃할 때 Netlet 종료 옵션 설정

사용자가 Portal Server 에서 로그아웃할 때 모든 연결이 종료되도록 하려는 경우에 이 옵션을 활성화합니다 . 이 옵션을 설정하면 보안이 강화됩니다 . 이 필드는 기본적으로 선택됩니다 .

사용자가 Portal Server 데스크탑을 로그아웃한 후에도 Netlet 연결이 계속 유지되도록 하려면 이 옵션을 비활성화합니다 .

참고 이 옵션을 비활성화시키면 사용자가 Portal Server 를 로그아웃한 후에 새 Netlet 연결을 구성할 수 없습니다 . 기존 연결만 보존됩니다 .

➤ **[포털 로그아웃할 때 Netlet 종료] 옵션을 설정하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .

3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [포털 로그아웃할 때 Netlet 종료] 필드로 스크롤하고 필요에 따라 옵션을 선택 또는 선택 해제합니다 .
8. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

   Sun Ray 환경에서 Netlet 실행을 참조하십시오 .

Netlet 규칙에 대한 액세스 정의

특정 조직 , 역할 또는 사용자의 특정 Netlet 규칙에 대한 액세스를 정의할 수 있습니다 .

➤ **Netlet 규칙에 대한 액세스를 정의하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [Netlet 규칙에 액세스] 필드로 스크롤합니다 .
8. [Netlet 규칙에 액세스] 필드에 선택 조직에서 이용할 수 있도록 할 규칙 이름을 입력합니다 .

   이 필드에서 별표 (*) 는 선택된 조직에 정의된 모든 Netlet 규칙을 사용할 수 있다는 것을 나타냅니다 .

9. [추가] 를 누릅니다 .

   지정한 규칙이 [Netlet 규칙에 액세스] 목록에 추가됩니다 .

10. 사용할 수 있도록 할 각 Netlet 규칙에 대해 단계 7, 8 및 9 를 반복합니다 .

11. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

Netlet 규칙에 액세스 거부

특정 조직 , 역할 또는 사용자의 특정 Netlet 규칙에 대한 액세스를 거부할 수 있습니다 .

➤ **Netlet 규칙에 액세스를 거부하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .

7. [Netlet 규칙 거부] 필드로 스크롤합니다 .
8. [Netlet 규칙 거부] 필드에 선택 조직에 대해 액세스를 거부할 규칙 이름을 입력합니다 .

   이 필드에서 별표 (*) 는 선택된 조직에 정의된 모든 Netlet 규칙이 거부된 액세스라는 것을 나타냅니다 .

9. [추가] 를 누릅니다 .

   지정한 규칙이 [Netlet 규칙 거부] 목록에 추가됩니다 .

10. 액세스를 거부할 각 Netlet 규칙에 대해 단계 7, 8 및 9 를 반복합니다 .
11. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

호스트에 대한 액세스 허용

특정 조직 , 역할 또는 사용자의 특정 호스트에 대한 액세스를 정의할 수 있습니다 . 그러면 특정 호스트에 대한 액세스를 허용할 수 있습니다 . 예를 들어 , 사용자가 텔넷으로 연결할 수 있는 5 개의 호스트로 허용 목록을 설정할 수 있습니다 .

➤ **호스트에 액세스를 허용하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [허용된 호스트] 필드로 스크롤합니다 .
8. [허용된 호스트] 필드에 액세스를 허용할 호스트 이름을 입력합니다 .

   이 필드에서 별표 (*) 는 지정 도메인의 모든 호스트에 액세스할 수 있다는 것을 나타냅니다 . 예를 들어 , `*.sesta.com` 을 지정하면 사용자가 `sesta.com` 도메인 내의 모든 Netlet 대상을 실행시킬 수 있습니다 . `xxx.xxx.xxx.*` 와 같이 IP 주소를 와일드 카드로 지정할 수도 있습니다 .
9. [추가] 를 누릅니다 .

   지정된 호스트가 [허용된 호스트] 목록에 추가됩니다 .
10. 사용할 수 있도록 할 각 호스트에 단계 7 및 8 을 반복합니다 .
11. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

호스트에 대한 액세스 거부

조직 내에서 특정 호스트에 대한 액세스를 거부할 수 있습니다 . [거부된 호스트] 목록에서 액세스를 거부할 호스트를 지정합니다 .

➤ **호스트에 액세스를 거부하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .

2. [Identity 관리] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. [보기] 드롭다운 목록에서 [서비스] 를 선택합니다 .
6. SRA 구성 아래에서 Netlet 옆에 있는 화살표를 클릭합니다 .

   오른쪽 창에 Netlet 페이지가 표시됩니다 .
7. [거부된 호스트] 필드로 스크롤합니다 .
8. [거부된 호스트] 필드에 액세스를 거부할 호스트 이름을 입력합니다 .

   이 필드에서 별표 (*) 는 선택된 조직 내의 모든 호스트에 사용자의 액세스가 거부된다는 것을 나타냅니다 . 예를 들어 , `sesta` 의 모든 호스트에 대한 액세스를 거부하려면 [거부된 호스트] 필드에 `*.sesta.com` 을 입력합니다 .

   특정 호스트에 액세스를 거부하려면 완전한 정규 이름을 지정합니다 . 예를 들어 , `abc` 호스트에 대한 액세스를 거부하려면 `abc.sesta.com` 을 입력합니다 .
9. [추가] 를 누릅니다 .

   지정한 도메인이 [도메인에 액세스] 목록에 추가됩니다 .
10. 사용할 수 있도록 할 각 도메인에 단계 7 및 8 을 반복합니다 .
11. Netlet 페이지 맨 위 또는 맨 아래의 [저장] 을 눌러 변경 사항을 저장합니다 .

프록시 구성

사용자 수준에서 다음과 같은 속성을 구성할 수 있습니다 .

- 브라우저 프록시 유형
- 브라우저 프록시 호스트
- 브라우저 프록시 포트
- 브라우저 프록시 대체 목록

관리 콘솔에서 이 값을 지정하지 않은 경우 Netlet 에서 브라우저 프록시 설정을 확인할 수 없으면 사용자가 처음으로 Netlet 을 통해 연결할 때 이 정보를 묻습니다 . 이 정보는 저장되었다가 나중에 사용자 연결에 사용됩니다 .

다음 시나리오에서는 Netlet 이 브라우저 프록시 설정을 확인하지 못합니다 .

- 사용자가 Java 플러그인 (버전 1.4.0 보다 낮음) 이 설치된 Internet Explorer 4.x, 5.x 또는 6.x 를 사용하고 , Java 플러그인 제어 패널의 [프록시] 탭에서 " 브라우저 설정 사용 " 옵션을 선택하고 , Internet Explorer 의 LAN 설정 대화 상자에 있는 " 자동 구성 스크립트 사용 " 필드에서 애드온 제품이나 INS 파일을 지정한 경우 .
- 사용자가 Java 플러그인 (버전 1.3.1_01 이상) 이 설치된 Netscape 6.2 를 사용하고 Java 플러그인 제어 패널의 [프록시] 탭에서 “브라우저 설정 사용” 옵션을 선택한 경우 .

두 경우 모두 , Netlet 이 브라우저 설정을 확인할 수 없기 때문에 사용자에게 다음 정보를 제공하도록 요청합니다 .

- 브라우저 프록시 유형

  이 속성은 DIRECT 또는 MANUAL 값을 가질 수 있습니다 . 사용자가 드롭다운 목록에서 DIRECT 를 선택하면 Netlet 이 게이트웨이 호스트에 직접 연결합니다 .

- 브라우저 프록시 호스트

  Netlet 의 연결을 위해 필요한 프록시 호스트를 지정합니다 .

- 브라우저 프록시 포트

  Netlet 의 연결을 위해 필요한 프록시 호스트의 포트를 지정합니다 .

- 브라우저 프록시 대체 목록 (쉼표로 구분)

  Netlet 이 프록시를 통해 연결하지 않도록 할 호스트를 지정합니다 . 이 목록에는 쉼표로 구분된 여러 호스트 이름이 있을 수 있습니다 .

디버그 로깅 활성화

디버그 정보의 위치는 Portal Server 노드의 `AMConfig-`*`instance-name`*`.properties` 파일에 있는 `com.iplanet.services.debug.directory` 속성 설정에 따라 결정됩니다 .

예를 들어 , `com.iplanet.services.debug.directory` 속성의 값이 다음과 같으면

`/var/opt/SUNWam/debug/`

NetFile 에 대한 디버그 정보는 `/var/opt/SUNWam/debug` 디렉토리에 있는 `srapNetlet` 파일에서 찾을 수 있습니다 .

자세한 내용은 *Identity Server 관리 설명서*를 참조하십시오 .

Proxylet 구성

이 장에서는 Sun Java™ System Identity Server 관리 콘솔에서 Proxylet 을 구성하는 방법에 대해 설명합니다 .

참고 Identity Server 관리 콘솔의 맨 위 오른쪽 구석에서 [도움말] 을 누르고 SRA 도움말을 눌러 모든 SRA 속성을 빠르게 참조할 수 있습니다 .

Proxylet 구성

Proxylet 채널 편집 페이지에서 [Proxylet 애플릿 자동으로 다운로드] 를 선택하면 사용자가 로그인할 때 자동으로 Proxylet 이 시작하도록 구성할 수 있습니다 . [Proxylet 자동으로 다운로드] 확인란을 선택하지 않은 경우 , 사용자는 표준 포털 데스크탑의 Proxylet 채널에 있는 "Proxylet 시작 " 링크를 눌러 필요할 때마다 Proxylet 을 가져올 수 있습니다 .

➤ **Proxylet 속성을 구성하려면**

1. Identity Server 관리 콘솔에 관리자로 로그인합니다 .
2. [서비스 구성] 탭을 선택합니다 .
3. [보기] 드롭다운 목록에서 [조직] 을 선택합니다 .
4. 필요한 조직 이름을 클릭합니다 . 선택된 조직 이름이 관리 콘솔의 상단 왼쪽 구석에 위치로 반영됩니다 .
5. SRA 구성 아래에서 Proxylet 옆에 있는 화살표를 클릭합니다 .
6. 원하는 경우 [Proxylet 애플릿 자동으로 다운로드] 확인란을 누릅니다 .
7. Proxylet 을 실행할 기본 Proxylet 애플릿 바인딩 IP 주소를 입력합니다 .

8. [기본 Proxylet 애플릿 포트] 필드에서 Proxylet 이 수신하는 포트 번호를 입력합니다 .

9. [저장] 을 누릅니다 .

참고 사용자가 Portal Server 에 로그인하여 Proxylet 을 실행한 다음 Java 플러그인을 설치한 경우에는 Netscape 브라우저를 다시 시작해야 합니다 .

SSL 가속기 구성

이 장에서는 Sun Java™ System Portal Server Secure Remote Access 에 다양한 가속기를 구성하는 방법을 설명합니다 .

이번 장에서는 다음 주제를 다룹니다 .

- Sun Crypto Accelerator 1000
- Sun Crypto Accelerator 4000
- 외부 SSL 장치 및 프록시 가속기

개요

외부 가속기는 서버 CPU 의 SSL 기능을 분담함으로써 CPU 가 다른 작업을 수행하도록 하여 SSL 트랜잭션의 처리 속도를 높이는 전용 하드웨어 코프로세서입니다 .

Sun Crypto Accelerator 1000

Sun™ Crypto Accelerator 1000 (Sun CA1000) 보드는 암호화 코프로세서로 작동하여 공용 키와 대칭 암호화를 가속화하는 짧은 형태의 PCI 보드입니다 . 이 제품에는 외부 인터페이스가 없습니다 . 이 보드는 내부 PCI 버스 인터페이스를 통해 호스트와 통신합니다 . 이 보드는 eCommerce 응용프로그램에서 보안 프로토콜을 위한 다양한 계산 집약적 암호화 알고리즘을 가속화하기 위한 목적으로 사용됩니다 .

RSA [7] 및 Triple-DES (3DES) [8] 와 같은 다수의 핵심 암호화 기능을 응용프로그램에서 Sun CA1000 으로 분담시켜 병렬로 수행할 수 있습니다 . 그러면 CPU 가 자유롭게 다른 작업을 수행할 수 있어 SSL 트랜잭션의 처리 속도가 증가합니다 .

Crypto Accelerator 1000 사용

Portal Server Secure Remote Access 가 설치되어 있고 게이트웨이 서버 인증서 (직접 서명 또는 CA 에서 발행) 가 설치되었는지 확인합니다 . 자세한 내용은 인증서 장을 참조하십시오 .

표 13-1 은 SSL 가속기를 설치하기 전에 필요한 정보를 추적하는 일을 돕는 점검 목록이며 Crypto Accelerator 1000 매개 변수와 값을 나열합니다 .

표 13-1 Crypto Accelerator 1000 설치 점검 목록

매개 변수	값
SRA 설치 기본 디렉토리	/opt
SRA 인증서 데이터베이스 경로	/etc/opt/SUNWps/cert/default
SRA 서버 인증서 별명	server-cert
영역	sra-keystore
영역 사용자	crypta

Crypto Accelerator 1000 구성

➤ **Crypto Accelerator 1000 을 구성하려면**

1. 사용 설명서의 지침에 따라 하드웨어를 설치합니다 . 참조 :

   `http://www.sun.com/products-n-solutions/hardware/docs/pdf/816-2450-11.pdf`

2. CD 에서 다음 패키지를 설치합니다 .

   `SUNWcrypm, SUNWcrypu, SUNWcrysu, SUNWdcar, SUNWcrypr, SUNWcrysl, SUNWdcamn, SUNWdcav`

3. 다음 패치를 설치합니다 . (`http://sunsolve.sun.com` 에서 얻을 수 있습니다 .)

   `110383-01, 108528-05, 112438-01`

4. pk12util 및 modutil 도구가 있는지 확인하십시오 .

   이 도구는 /usr/sfw/bin 아래에 설치됩니다 . /usf/sfw/bin 디렉토리에 도구가 없는 경우에는 Sun Java System 배포 매체에서 SUNWtlsu 패키지를 수동으로 추가해야 합니다 .

   ```
   Solaris_[sparc/x86]/Product/shared_components/
   ```

5. 슬롯 파일을 만듭니다 .

   ```
   vi /etc/opt/SUNWconn/crypto/slots
   ```

   그리고 파일의 처음이자 유일한 라인으로 "crypta@sra" 를 넣습니다 .

6. 영역을 만들고 설정합니다 .

   a. 루트로 로그인합니다 .

   b. 다음 명령을 입력합니다 .

      ```
      cd /opt/SUNWconn/bin/secadm
      secadm> create realm=sra
      ```

      영역 sra 가 성공적으로 만들어졌습니다 .

7. 사용자를 만듭니다 .

   a. 다음 명령을 입력하고 응답합니다 .

      secadm> set realm=sra

      secadm{srap}> su

      secadm{root@sra}>create user=crypta

      초기 비밀번호 :

      비밀번호 확인 :

      사용자 crypta 가 성공적으로 만들어졌습니다 .

8. 만든 사용자로 로그인합니다 .

   secadm{root@sra}> login user=crypta

   비밀번호 :

   secadm{crypta@sra}> show key

   이 사용자에게 키가 없습니다 .

9. Sun Crypto 모듈을 로드합니다 .

   환경 변수 LD_LIBRARY_PATH 는 /usr/lib/mps/secv1/ 을 가리켜야 합니다 .

   다음을 입력합니다 .

   ```
   modutil -dbdir /etc/opt/SUNWps/cert/default -add "Sun Crypto Module"
   -libfile /opt/SUNWconn/crypto/lib/libpkcs11.so
   ```

   다음 명령을 사용하여 이 모듈이 로드되었는지 확인합니다 .

   ```
   modutil -list -dbdir /etc/opt/SUNWps/cert /default
   ```

10. 게이트웨이 인증서와 키를 "Sun Crypto Module" 로 내보냅니다 .

    환경 변수 LD_LIBRARY_PATH 는 /usr/lib/mps/secv1/ 을 가리켜야 합니다 .

    다음을 입력합니다 .

    ```
    pk12util -o servercert.p12 -d /etc/opt/SUNWps/cert/default -n
    server-cert
    ```

    ```
    pk12util -i servercert.p12 -d /etc/opt/SUNWps/cert/default -h
    "crypta@sra"
    ```

    이제 show key 명령을 실행합니다 .

    secadm{crypta@sra}> show key

    이 사용자에게 2 개의 키가 나타나야 합니다 .

11. /etc/opt/SUNWps/cert/default/.nickname 파일에서 별명을 변경합니다 .

    ```
    vi /etc/opt/SUNWps/cert/default/.nickname
    ```

    server-cert 를 crypta@sra:server-cert 로 교체합니다 .

12. 가속화용 암호를 활성화합니다 .

    266 페이지의 "SSL 암호 선택 사용 " 을 참조하십시오 .

    SUN CA1000 은 RSA 기능을 가속화하지만 DES 와 3DES 암호에 대한 가속만 지원합니다 .

13. 가속기를 사용하도록 /etc/opt/SUNWps/platform.conf.*gateway-profile-name* 을 수정합니다 .

    ```
    gateway.enable.accelerator=true
    ```

14. 단말기 창에서 게이트웨이를 다시 시작합니다 .

    portal-server-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

참고 게이트웨이는 게이트웨이 프로필에서 https 포트로 언급된 포트의 일반 ServerSocket(비 SSL) 에 바인딩합니다 .

들어오는 클라이언트 트래픽에 대해 SSL 암호화 또는 복호화가 수행되지 않습니다 . 가속기에서 이 작업을 수행합니다 .

PDC 는 이 모드에서 작동하지 않습니다 .

Sun Crypto Accelerator 4000

Sun™ Crypto Accelerator 4000 보드는 Sun 서버에서 IPsec 및 SSL(대칭 및 비대칭 모두) 에 대한 암호화 하드웨어 가속을 지원하는 기가비트 이더넷 기반 네트워크 인터페이스 카드입니다 .

암호화되지 않은 네트워크 트래픽을 위한 표준 기가비트 이더넷 네트워크 카드로 작동하는 외에 이 보드에는 암호화 IPsec 트래픽에 높은 처리 속도를 지원할 암호 하드웨어가 포함되어 있습니다 .

Crypto Accelerator 4000 보드는 하드웨어와 소프트웨어 모두에서 암호화 알고리즘을 가속화합니다 . 암호 DES 및 3DES 에 대한 대량 암호화도 지원합니다 .

Crypto Accelerator 4000 사용

SRA 가 설치되어 있고 게이트웨이 서버 인증서 (직접 서명 또는 CA 에서 발행) 가 설치되었는지 확인합니다 . 다음 점검 목록으로 SSL 가속기를 설치하기 전에 필요한 정보를 쉽게 확인할 수 있습니다 .

표 13-2 은 Crypto Accelerator 4000 매개 변수와 그 값을 나열합니다 .

표 13-2 Crypto Accelerator 4000 설치 점검 목록

매개 변수	값
Secure Remote Access 설치 기본 디렉토리	/opt
SRA 인스턴스	기본값
SRA 인증서 데이터베이스 경로	/etc/opt/SUNWps/cert/default
SRA 서버 인증서 별명	server-cert
CA4000 키 저장소	srap

표 13-2 Crypto Accelerator 4000 설치 점검 목록

매개 변수	값
CA4000 키 저장소 사용자	crypta

Crypto Accelerator 4000 구성

➤ **Crypto Accelerator 4000 을 구성하려면**

1. 사용 설명서의 지침에 따라 하드웨어와 소프트웨어 패키지를 설치합니다 . 참조 :

   http://www.sun.com/products-n-solutions/hardware/docs/pdf/816-2450-11.pdf

2. 다음 패치를 설치합니다 . (http://sunsolve.sun.com 에서 얻을 수 있습니다 .)
   114795

3. certutil, pk12util 및 modutil 도구가 있는지 확인하십시오 .

   이 도구는 /usr/sfw/bin 아래에 설치됩니다 .

   /usf/sfw/bin 디렉토리에서 도구를 사용할 수 없는 경우에는

   Sun Java System 배포 매체에서 수동으로 SUNWtlsu 패키지를 추가해야 합니다 .

   Solaris_[sparc/x86]/Product/shared_components/

4. 보드를 초기화합니다 .

   /opt/SUNWconn/bin/vcadm 도구를 실행하여 암호화 보드를 초기화하고 다음 값을 설정합니다 .

   초기 보안 관리 이름 : sec_officer

   키 저장소 이름 : sra-keystore

   FIPS 140-2 모드에서 실행 : No

5. 사용자를 만듭니다 .

   vcaadm{vca0@localhost, sec_officer}> create user

   새 사용자 이름 : crypta

   새 사용자 비밀번호 입력 :

   비밀번호 확인 :

   사용자 crypta 가 성공적으로 만들어졌습니다 .

6. 키 저장소에 토큰을 매핑합니다 .

   vi /opt/SUNWconn/cryptov2/tokens

   그리고 파일에 sra-keystore 를 추가합니다 .

7. 대량 암호화의 사용을 설정합니다 .

   touch /opt/SUNWconn/cryptov2/sslreg

8. Sun Crypto 모듈을 로드합니다 .

   환경 변수 LD_LIBRARY_PATH 는 /usr/lib/mps/secv1/ 을 가리켜야 합니다 .

   다음을 입력합니다 .

   modutil -dbdir /etc/opt/SUNWps/cert/default -add "Sun Crypto Module" -libfile /opt/SUNWconn/cryptov2/lib/libvpkcs11.so

   다음 명령을 사용하여 이 모듈이 로드되었는지 확인할 수 있습니다 .

   modutil -list -dbdir /etc/opt/SUNWps/cert /default

9. 게이트웨이 인증서와 키를 "Sun Crypto Module" 로 내보냅니다 .

   환경 변수 LD_LIBRARY_PATH 는 /usr/lib/mps/secv1/ 을 가리켜야 합니다 .

   pk12util -o servercert.p12 -d /etc/opt/SUNWps/cert/default -n server-cert

   pk12util -i servercert.p12 -d /etc/opt/SUNWps/cert/default -h "sra-keystore"

   다음 명령을 사용하여 키가 내보내졌는지 확인할 수 있습니다 .

   certutil -K -h "sra-keystore" -d /etc/opt/SUNWps/cert/default

10. /etc/opt/SUNWps/cert/default/.nickname 파일에서 별명을 변경합니다 .

    vi /etc/opt/SUNWps/cert/default/.nickname

    server-cert 를 sra-keystore:server-cert 로 교체합니다 .

11. 가속화용 암호를 활성화합니다 .

    266 페이지의 "SSL 암호 선택 사용 " 을 참조하십시오 .

12. 단말기 창에서 게이트웨이를 다시 시작합니다 .

portal-server-install-root/SUNWps/bin/gateway -n *gateway-profile-name* start

게이트웨이가 키 저장소 비밀번호를 입력하도록 요청합니다 .

"sra-keystore":crypta:crytpa-password 에 대한 비밀번호 또는 PIN 을 입력합니다 .

참고 게이트웨이는 게이트웨이 프로필에서 https 포트로 언급된 포트의 일반 ServerSocket(비 SSL) 에 바인딩합니다 .

들어오는 클라이언트 트래픽에 대해 SSL 암호화 또는 복호화가 수행되지 않습니다 . 가속기에서 이 작업을 수행합니다 .

PDC 는 이 모드에서 작동하지 않습니다 .

외부 SSL 장치 및 프록시 가속기

열린 모드에서 외부 SSL 장치를 Sun Java System Portal Server Secure Remote Access (SRA) 전방에서 실행할 수 있습니다 . 이 장치는 클라이언트와 SRA 사이에서 SSL 링크를 제공합니다 .

외부 SSL 장치 가속기 사용

SRA 가 설치되어 있고 게이트웨이가 보안 모드 (HTTPS 모드) 에서 실행되는지 확인합니다 .

게이트웨이 >> HTTPS 연결 사용

게이트웨이 >> HTTP 포트 : 880

표 13-3 은 외부 SSL 장치와 프록시 가속기 매개 변수 및 값을 나타냅니다 .

표 13-3 외부 SSL 장치 및 프록시 가속기 점검 목록

매개 변수	값
SRA 인스턴스	기본값
게이트웨이 모드	https
게이트웨이 포트	880
외부 장치 / 프록시 포트	443

외부 SSL 장치 가속기 구성

➤ 외부 SSL 장치 가속기를 구성하려면

1. 사용 설명서의 지침에 따라 하드웨어와 소프트웨어 패키지를 설치합니다 .
2. 해당하는 경우 필요한 패치를 설치합니다 .
3. `platform.conf` 파일에 값을 입력하여 SSL 장치 / 프록시 지원을 활성화합니다 .

   ```
   vi /etc/opt/SUNWps/platform.conf.default

   gateway.enable.accelerator=true
   ```

   외부 장치 / 프록시 호스트 이름이 게이트웨이 호스트 이름과 다른 경우 :

   ```
   gateway.enable.customurl=true

   gateway.httpsurl=external-device.domain.subdomain/proxy-URL
   ```

4. 두 가지 방법으로 게이트웨이 알림을 구성할 수 있습니다 .
   - Identity Server 가 포트 880 에서 게이트웨이 컴퓨터와 접속할 수 있는 경우 (http 로 세션 알림) `platform.conf` 파일에 값을 입력합니다 .

     ```
     vi /etc/opt/SUNWps/platform.conf.default

     gateway.protocol=http

     gateway.port=880
     ```

   - Identity Server 가 포트 443 에서 외부 장치 / 프록시와 접속할 수 있는 경우 (HTTPS 세션 알림) `platform.conf` 파일에 값을 입력합니다 .

     ```
     vi /etc/opt/SUNWps/platform.conf.default

     gateway.host=External Device/Proxy Host Name

     gateway.protocol=https

     gateway.port=443
     ```

5. SSL 장치 / 프록시가 작동하고 있으며 게이트웨이 포트로 트래픽을 넘기도록 구성되어 있는지 확인합니다 .
6. 단말기 창에서 게이트웨이를 다시 시작합니다 .

   gateway-install-root`/SUNWps/bin/gateway -n` *gateway-profile-name* `start`

부록 A

로그 파일

다음 로그 파일은 기본 /var/opt/SUNWps/debug 디렉토리에 있으며 디버그 및 기타 유형의 정보를 포함합니다 .

표 A-1 정보 및 디버그 파일

파일 이름	목차
다음 로그 파일은 기본 디렉토리 /etc/opt/SUNWam/config/ 파일의 AMConfig-*instance-name*.properties 파일에 있는 디버그 매개 변수로 제어됩니다 .	
amconsole	Netfile, Netlet 및 게이트웨이 관리 파일
srapNetFile	NetFile 정보 파일
srapNetlet	Netlet 정보 파일
다음 로그 파일은 기본 디렉토리 /etc/opt/SUNWps: 에 있는 platform.conf.*gateway-profile-name* 파일의디버그 매개 변수 gateway.debug 로 제어됩니다 .	
srapGateway.*gateway-profile-name*	게이트웨이 정보
Gateway_to_from_server.*gateway-profile-name*	
Gateway_to_from_browser.*gateway-profile-name*	
srapNetletProxy.*gateway-profile-name*	
srapRewriterProxy.*gateway-profile-name*	
rwproxy.log.*rewriter-proxy-instance-name*	Rewriter 프록시의 시작 및 중지 시간
nlproxy.log.*netlet-proxy-instance-name*	Netlet 프록시의 시작 및 중지 시간
gateway.log.*gateway.instance.name*	게이트웨이의 시작 및 중지 시간
다음 Rewriter 파일은 기본 디렉토리 /var/opt/SUNWam/config/ 파일에 있는 AMConfig-*instance-name*.properties 파일의 디버그 매개 변수로 제어합니다 . 자세한 내용은 135 페이지의 " 디버깅 로그 사용의 문제 해결 " 을 참조하십시오 .	
RuleSetInfo	재작성에 사용된 모든 규칙 집합이 이 파일에 기록됩니다 .

표 A-1 정보 및 디버그 파일

파일 이름	목차
Original Pages	페이지 URI, resolveURI(페이지 URI 와 다른 경우), 컨텐트 MIME, 페이지에 적용된 규칙 집합 , 구문 분석기 MIMIE 및 원본 컨텐트가 들어 있습니다 . 구문 분석과 관련된 특정 오류 / 경고 / 메시지도 이 파일에 들어 있습니다 . 메시지 모드에서는 전체 컨텐트가 기록되고 경고와 오류 모드에서는 재작성 중에 발생한 예외만 기록됩니다 .
Rewritten Pages	페이지 URI, resolveURI(페이지 URI 와 다른 경우), 컨텐트 MIME, 페이지에 적용된 규칙 집합 , 구문 분석기 MIMIE 및 재작성된 컨텐트가 들어 있습니다 . 이 파일은 디버그 모드가 메시지로 설정되었을 때 채워집니다 .
Unaffected Pages	수정되지 않은 페이지 목록을 포함합니다 .
URIInfo Pages	이 파일에는 발견되어 변환된 URL 이 들어 있습니다 . 컨텐트가 원본 데이터와 동일하게 유지되는 모든 페이지의 세부 사항이 이 파일에 기록됩니다 . 세부적으로 기록되는 내용 : 페이지 URI, MIME 및 인코딩 데이터 , 재작성에 사용된 rulesetID 그리고 구문 분석기 MIME.

부록 B

구성 속성

이 부록에서는 각 Secure Remote Access 구성 요소의 서비스 구성에서 Identity Server 관리 콘솔을 통해 Portal Server Secure Remote Access에 대해 구성할 수 있는 속성을 설명합니다 .

- 액세스 목록 서비스
- 게이트웨이 서비스
- NetFile 서비스
- Netlet 서비스
- Proxylet 서비스

액세스 목록 서비스

표 B-1 에는 액세스 목록 서비스 속성이 나열되어 있습니다 .

표 B-1 액세스 목록 서비스 속성

속성	기본값	설명
거부된 URL		최종 사용자가 게이트웨이를 통해 액세스할 수 없는 URL 목록
허용된 URL	*	최종 사용자가 게이트웨이를 통해 액세스할 수 없는 URL 목록
단일 사인온이 금지된 호스트		호스트 목록에 대해 단일 사인온 사용을 해제합니다 .
세션별 단일 사인온 허용		세션에 대해 단일 사인온의 사용을 설정합니다 .
허용된 인증 수준	*	인증을 어느 정도나 신뢰할지를 나타냅니다 . 모든 인증 수준을 허용하려면 별표를 사용합니다 . 인증 수준에 대한 내용은 *Identity Server 관리 설명서*를 참조하십시오 .

게이트웨이 서비스

게이트웨이 서비스를 누르면 오른쪽 표시 영역에 새 프로필을 만들기 위한 버튼과 만든 게이트웨이 프로필 목록이 표시됩니다 .

[새로 만들기] 를 누르면 다음 표시 영역에 새 게이트웨이 프로필 이름을 입력하라는 메시지가 표시됩니다 . 기본 템플리트 또는 이전에 만든 게이트웨이 프로필을 템플리트로 사용할 수 있습니다 .

나열된 게이트웨이 프로필 이름 중 하나를 누르면 탭 목록이 제시됩니다 . 다음 탭이 있습니다 .

- 핵심
- 프록시
- 보안
- Rewriter
- 기록

핵심

표 B-2 에는 게이트웨이 서비스 핵심 속성이 나와 있습니다 .

표 B-2 게이트웨이 서비스 코어 속성

속성	기본값	설명
HTTPS 연결 사용		HTTPS 연결의 사용을 설정합니다 .
HTTPS 포트	443	HTTPS 포트를 지정합니다 .
HTTP 연결 사용	*	HTTP 연결의 사용을 설정합니다 .
HTTP 포트	80	HTTP 포트를 지정합니다 .
Rewriter 프록시 사용	*	게이트웨이와 인트라넷 사이에서 보안 HTTP 트래픽의 사용을 설정합니다 . Rewriter 프록시와 게이트웨이는 같은 게이트웨이 프로필을 사용합니다 .
Rewriter 프록시 목록		Rewriter 프록시 목록 . Rewriter 프록시 인스턴스가 여럿인 경우에는 각각 *host-name:port* 의 형식으로 세부 사항을 입력합니다 .
Netlet 사용	선택	TCP/IP (텔넷 및 SMTP 등), HTTP 응용프로그램 및 고정 포트 응용프로그램을 위한 보안의 사용을 설정합니다 .

표 B-2 게이트웨이 서비스 코어 속성

속성	기본값	설명
Proxylet 사용	선택	클라이언트 컴퓨터에서 Proxylet 의 다운로드를 허용합니다 .
Netlet 프록시 사용		클라이언트로부터 게이트웨이를 거쳐 인트라넷에 상주하는 Netlet 프록시까지 보안 터널을 확장함으로써 게이트웨이와 인트라넷 사이의 Netlet 트래픽 보안을 강화합니다 . Portal Server 에서 응용프로그램을 사용하지 않으려면 사용 해제합니다 .
Netlet 프록시 호스트		host hostname:port 형식으로 Netlet 프록시 호스트를 나열합니다 .
쿠키 관리 사용		사용자가 액세스할 수 있는 모든 웹 사이트의 사용자 세션을 추적하고 관리합니다 . (Portal Server 에서 Portal Server 사용자 세션을 추적하기 위해 사용하는 쿠키에는 적용되지 않습니다 .
HTTP 기본 인증 사용	선택	BASIC 으로 보호된 웹 사이트를 다시 방문할 때 자격 증명 정보를 다시 입력하지 않도록 아이디와 비밀번호를 저장할 수 있습니다 .
HTTP 지속 연결 사용	선택	게이트웨이에서 HTTP 지속 연결을 사용하여 웹 페이지의 모든 개체 (이미지 및 스타일 시트 등) 에 대해 소켓이 열리는 것을 방지합니다 .
지속 연결당 최대 요청 수	10	지속 연결당 요청 수를 지정합니다 .
지속적 소켓 연결 시간 초과	50	소켓이 닫힐 때까지 경과해야 하는 시간을 지정합니다 .
반환 시간을 위한 계정의 유예 시간 초과	20	브라우저가 요청을 보낸 후 요청이 게이트웨이에 도달하기 위한 유예 시간과 응답을 보내는 게이트웨이와 실제로 이를 받는 브라우저 사이의 시간을 지정합니다 .
사용자 세션 쿠키가 전달될 URL		서블릿 및 CGI 가 Portal Server 의 쿠키를 수신하고 API 를 사용하여 사용자를 확인하도록 합니다 .
최대 연결 대기 길이	50	게이트웨이가 허용할 수 있는 최대 동시 연결을 지정합니다 .
게이트웨이 시간 초과 (밀리초)	120000	게이트웨이가 브라우저와의 연결에서 시간 초과할 시간을 밀리초 단위로 지정합니다 .
최대 스레드 풀 크기	200	게이트웨이 스레드 풀에서 사전에 생성할 수 있는 최대 스레드 수를 지정합니다 .
캐시된 소켓 시간 초과	200000	게이트웨이가 Portal Server 와의 연결에서 시간 초과할 시간을 밀리초 단위로 지정합니다 .

표 B-2 게이트웨이 서비스 코어 속성

속성	기본값	설명
Portal Server		`http://`*portal server name:port -number* 형식으로 Portal Server 를 지정합니다 . 게이트웨이는 요청을 처리하기 위해 연속해서 나열된 각 Portal Server 에 접속을 시도합니다 .
서버 재시도 간격 (분)	2	Portal Server, Rewriter 프록시 또는 Netlet 프록시를 사용할 수 없게 된 후 (충돌이나 다운된 경우) 시작 요청 사이의 시간 간격을 지정합니다 .
외부 서버 쿠키 저장		게이트웨이에서 게이트웨이를 통해 액세스하는 타사 응용프로그램이나 서버에 대한 쿠키를 저장하고 관리할 수 있습니다 .
URL 에서 세션 정보 얻기		쿠키가 지원되는지 여부에 상관 없이 세션 정보를 URL 의 일부로 코드화합니다 . 게이트웨이는 클라이언트의 브라우저에서 보내는 세션 쿠키를 사용하지 않고 검증을 위해 URL 에 있는 이 세션 정보를 사용합니다 .
쿠키를 안전하다고 표시		쿠키를 안전하다고 표시합니다 . [쿠키 관리 사용] 옵션을 선택해야 합니다 .

프록시

표 B-3 에는 게이트웨이 서비스 프록시 속성이 나와 있습니다 .

표 B-3 게이트웨이 서비스 프록시 속성

속성	기본값	설명
프록시 사용		웹 프록시의 사용을 설정합니다 .
웹 프록시 URL 사용		프록시 사용 옵션이 선택되지 않아도 게이트웨이가 [도메인 및 부속 도메인 프록시] 목록에 나열된 웹 프록시를 통해서만 접속해야 하는 URL 을 나열합니다 .
웹 프록시 URL 사용 안함		게이트웨이가 직접 연결할 수 있는 URL 을 나열합니다 .
도메인 및 부속 도메인의 프록시	iportal.com sun.com	특정 도메인의 특정 부속 도메인에 접속하기 위해 사용할 프록시를 지정합니다 .
프록시 비밀번호 목록		프록시 서버가 일부 또는 모든 사이트에 대한 액세스에 인증을 요구하는 경우 게이트웨이가 지정된 프록시 서버에 인증을 얻기 위해 필요한 사용자 이름과 비밀번호를 지정합니다 .

표 B-3 게이트웨이 서비스 프록시 속성

속성	기본값	설명
자동 프록시 구성 지원 사용		[도메인 및 부속 도메인의 프록시] 필드에 제공된 정보가 무시되도록 지정합니다 .
자동 프록시 구성 파일 위치		PAC 지원에 사용할 파일 위치를 지정합니다 .
웹 프록시를 통한 Netlet 터널링 허용		클라이언트로부터의 보안 터널을 게이트웨이를 통해 인트라넷에 있는 웹 프록시로 연장합니다 .

보안

표 B-4 에는 게이트웨이 서비스 보안 속성이 나와 있습니다 .

표 B-4 게이트웨이 서비스 보안 속성

속성	기본값	설명
비인증 URL	/portal/desktop/images /amserver/login_images /portal/desktop/css /amserver/jss /amconsole/console/css /portal/searchadmin/console/js /amconsole/console/js /amserver/css	이미지가 있는 디렉토리와 같이 인증이 필요 없는 URL 을 지정합니다 .
인증서 사용 가능 게이트웨이 호스트		인증서 사용 가능 게이트웨이 호스트를 나열합니다 .
40 비트 암호화 허용		40 비트 (취약) SSL(Secure Sockets Layer) 연결을 허용합니다 . 이 옵션을 선택하지 않으면 128 비트 연결만 지원됩니다 .
SSL 버전 2.0 사용	선택	SSL 버전 2.0 의 사용을 설정합니다 . SSL 2.0 을 사용 해제하면 구식 SSL 2.0 만 지원하는 브라우저가 SRA 에 인증을 얻을 수 없게 됩니다 . 이렇게 하면 더 높은 수준의 보안이 보장됩니다 .
SSL 암호 선택 사용		SSL 암호 선택의 사용을 설정합니다 . 사전 구성된 모든 암호의 지원을 선택하거나 필요한 암호만 개별적으로 선택할 수 있습니다 . 각 게이트웨이 인스턴스에 특정 SSL 암호를 선택할 수 있습니다 .
SSL2 암호		선택할 수 있는 SSL 버전 2 암호를 나열합니다 .
SSL3 암호		선택할 수 있는 SSL 버전 3 암호를 나열합니다 .

표 B-4 게이트웨이 서비스 보안 속성

속성	기본값	설명
TLS 암호		TLS 암호를 나열합니다 .
SSL 버전 3.0 사용	선택	SSL 버전 3.0 의 사용을 설정합니다 . SSL 3.0 을 사용 해제하면 SSL 3.0 만 지원하는 브라우저가 SRA 에 인증을 얻을 수 없게 됩니다 . 그러면 보안 수준이 높아집니다 .
Null 암호 사용		Null 암호를 활성화합니다 .
인증된 SSL 도메인		인증된 SSL 도메인을 나열합니다 .

Rewriter

Rewriter 탭에는 두 개의 하위 부분이 있습니다 .

- 기본
- 고급

기본

표 B-5 에는 게이트웨이 서비스 Rewriter 기본 속성이 나와 있습니다 .

표 B-5 게이트웨이 서비스 Rewriter 속성 - 기본

속성	기본값	설명
모든 URL 다시 쓰기 사용		도메인 및 부속 도메인의 프록시 목록에 있는 항목을 점검하지 않고 모든 URL 이 다시 작성되도록 지정합니다 .

표 B-5 게이트웨이 서비스 Rewriter 속성 - 기본

속성	기본값	설명
규칙 집합에 URI 매핑	*://*.iportal.com*/portal/*|default_gateway_ruleset */portal/NetFileOpenFileServlet*|null_ruleset *|generic_ruleset REPLACE_WITH_IPLANET_MAIL_SERVER_NAME|iplanet_mail_ruleset REPLACE_WITH_EXCHANGE_SERVER_NAMEexchange_2000sp3_owa_ruleset *://*.iportal.com*/amconsole/*|default_gateway_ruleset REPLACE_WITH_INOTES_SERVER_NAME|inotes_ruleset http*://*/portal/NetFileController*|null_ruleset	맵 URI 와 규칙 집합 목록을 사용하여 도메인을 규칙 집합과 연관시킵니다 . 규칙 집합은 Identity Server 관리 콘솔의 포털 서버 구성에서 만듭니다 .
구문 분석기를 MIME 유형에 매핑	JAVASCRIPT=application/x-java XML=text/xml HTML=text/html;text/htm;text/x-component;text/wml;text/vnd.wap.wml CSS=text/css	새 MIME 유형을 HTML, JAVASCRIPT, CSS 또는 XML 과 연관시킵니다 . 여러 항목을 입력할 때는 세미콜론이나 콤마로 구분합니다 .
다시 쓰지 않는 URI		다시 쓰지 않을 URI 를 나열합니다 . 참고 : 이 목록에 #* 를 추가하면 href 규칙이 규칙 집합의 일부라 하더라도 URI 를 다시 쓸 수 있습니다 .
기본 도메인		기본 도메인 및 부속 도메인에 대한 호스트 이름을 확인합니다 . 이 정보는 설치 중에 구성됩니다 .

고급

표 B-6 에는 게이트웨이 서비스 Rewriter 고급 속성이 나와 있습니다 .

표 B-6 게이트웨이 서비스 Rewriter 속성 ñ 고급

속성	기본값	설명
MIME 추측 사용		MIME 이 전송되지 않을 때 MIME 추측의 사용을 설정합니다 . 구분 분석기와 URI 의 매핑 목록 상자에 데이터를 추가해야 합니다 .
구문 분석기와 URI 의 매핑		구문 분석기를 URI 에 매핑합니다 . 각 URI 는 세미콜론으로 구분합니다 . 예 : HTML=*.html; *.htm;*Servlet 이 것은 html, htm 또는 Servlet 확장을 가진 모든 페이지에 대한 컨텐트를 다시 쓰기 위해 Rewriter 가 사용된다는 것을 의미합니다 .
마스크 사용		Rewriter 가 페이지의 인트라넷 URL 이 보이지 않도록 URI 를 다시 쓸 수 있습니다 .
마스킹용 씨드 문자열		URI 를 마스크하는 데 사용되는 씨드 문자열을 지정합니다 . 씨드 문자열은 마스킹 알고리즘에 의해 생성되는 임의 문자열입니다 .
마스크하지 않을 URI		마스크하지 않을 인터넷 URI 를 지정합니다 . 응용프로그램 (애플릿 등) 에 인터넷 URI 가 필요한 경우에 사용됩니다 . 예를 들어 다음을 목록 상자에 추가하면 */Applet/Param* 컨텐트 URI `http://abc.com/Applet/Param1.html` 이 규칙 집합의 규칙에서 매칭되는 경우 URL 이 마스크되지 않습니다 .
게이트웨이 프로토콜을 원본 URI 프로토콜과 같게 만들기		HTML 컨텐트의 참조 리소스에 액세스할 때 Rewriter 가 일관된 프로토콜을 사용하도록 합니다 . 이는 Javascript 로 생성된 동적 URI 가 아닌 정적 URI 에만 적용됩니다 .

기록

표 B-7 에는 게이트웨이 서비스 기록 속성이 나와 있습니다 .

표 B-7 게이트웨이 서비스 로깅 속성

속성	기본값	설명
기록 사용		기록의 사용을 설정합니다 .
세션 기록마다 사용		클라이언트 주소 , 요청 유형 및 대상 호스트와 같은 최소 기록 정보를 포착하도록 합니다 .
세션 기록마다 상세 정보 표시 사용		클라이언트 , 요청 유형 , 대상 호스트 , 요청의 유형 , 클라이언트 요청 URL, 클라이언트 사후 데이터 크기 , 세션 아이디 , 요청 결과 코드 및 전체 응답 크기와 같은 세부적 기록 정보를 포착하도록 합니다 . 참고 : 세션 기록마다 사용을 선택해야 합니다 .
Netlet 기록 사용		기록을 사용할지 여부를 지정합니다 . 기록을 사용하는 경우 , 다음 정보가 포착됩니다 . 시작 시간 , 소스 , 주소 , 소스 포트 , 서버 주소 , 서버 포트 , 중단 시간 , 상태 (시작 또는 중단)

NetFile 서비스

NetFile 서비스를 누르면 오른쪽 창에 탭이 표시됩니다 . 다음 탭이 있습니다 .

- 호스트
- 권한
- 보기
- 작업
- 일반

호스트

호스트 탭에는 두 개의 하위 부분이 있습니다 .

- 구성
- 액세스

구성

표 B-8 은 NetFile 호스트 구성 속성의 목록입니다 .

표 B-8 NetFile 서비스 호스트 구성 속성

속성	기본값	설명
OS 문자 집합	Unicode(UTF-8)	호스트와의 통신을 위한 기본 인코딩으로 사용할 문자 집합을 지정합니다 .
호스트 검색 순서	WIN, NETWARE, FTP, NFS	호스트 검색 순서를 지정합니다 .
공통 호스트		NetFile 을 통해 모든 원격 NetFile 사용자가 사용할 수 있게 만들 호스트를 지정합니다 .
기본 도메인		허용된 호스트에 접속하기 위해 NetFile 이 사용해야 하는 기본 도메인을 지정합니다 .
기본 Windows 도메인 / 워크그룹		사용자가 Windows 호스트에 액세스할 때 선택하는 기본 Windows 도메인 또는 워크그룹을 지정합니다 .
기본 WINS/DNS 서버		NetFile 이 Windows 호스트에 액세스할 때 사용하는 WINS/DNS 서버를 지정합니다 .

액세스

표 B-9 는 NetFile 서비스 호스트 액세스 속성의 목록입니다 .

표 B-9 NetFile 서비스 호스트 액세스 속성

속성	기본값	설명
Windows 호스트에 액세스 허용	선택	Windows 호스트에 액세스를 허용합니다 .
FTP 호스트에 액세스 허용	선택	FTP 호스트에 액세스를 허용합니다 .
NFS 호스트에 액세스 허용	선택	NFS 호스트에 액세스를 허용합니다 .
Netware 호스트에 액세스 허용	선택	Netware 호스트에 액세스를 허용합니다 .
허용된 호스트	*	사용자가 NetFile 을 통해 액세스할 수 있는 호스트를 지정합니다 .
거부된 호스트		사용자가 NetFile 을 통해 액세스할 수 없는 호스트를 지정합니다 .

권한

사용자가 NetFile 을 사용하기 시작한 후에 이러한 옵션을 사용 해제하면 사용자가 NetFile 을 로그아웃하고 다시 로그인하는 경우에만 변경 사항이 적용됩니다 .

표 B-10 은 NetFile 서비스 권한 속성의 목록입니다 .

표 B-10 NetFile 서비스 권한 속성

속성	기본값	설명
파일 이름 변경 허용	선택	사용자가 파일의 이름을 바꿀 수 있습니다 .
파일 / 폴더 삭제 허용	선택	사용자가 파일 및 폴더를 삭제할 수 있습니다 .
파일 업로드 허용	선택	사용자가 파일을 업로드할 수 있습니다 .
파일 / 폴더 다운로드 허용	선택	사용자가 파일 및 폴더를 다운로드할 수 있습니다 .
파일 검색 허용	선택	사용자가 검색할 수 있습니다 .
파일 메일 허용	선택	파일 메일링을 허용합니다 .
파일 압축 허용	선택	파일 압축을 허용합니다 .
사용자 아이디 변경 허용	선택	사용자가 다른 아이디를 사용할 수 있습니다 .
Windows 도메인 변경 허용	선택	사용자가 Windows 도메인을 변경할 수 있습니다 .

보기

표 B-11 은 NetFile 서비스 보기 속성의 목록입니다 .

표 B-11 NetFile 서비스 보기 속성

속성	기본값	설명
창 크기	700|400	사용자 데스크탑에서 픽셀 단위로 NetFile 창의 크기를 지정합니다 . 잘못된 값을 입력하면 NetFile 이 기본값을 사용합니다 .
창 위치	100|50	사용자 데스크탑에서 픽셀 단위로 NetFile 창의 크기를 지정합니다 . 잘못된 값을 입력하면 NetFile 이 기본값을 사용합니다 .

작업

작업 탭에는 다음 하위 부분이 있습니다 .

- 트래픽

- 검색
- 압축

트래픽

표 B-12 는 NetFile 서비스 작업 트래픽 속성의 목록입니다 .

표 B-12 NetFile 서비스 작업 - 트래픽 속성

속성	기본값	설명
임시 디렉토리 위치	/tmp	다양한 NetFile 파일 작업을 위한 임시 디렉토리를 지정합니다 . 웹 서버 실행에 사용하고 있는 아이디 (nobody 또는 noaccess 등) 에 지정 디렉토리에 대한 rwx 권한이 있는지 확인하십시오 . 이 아이디에 필요한 임시 디렉토리의 전체 경로에 대한 rx 권한이 있는지도 확인하십시오 . NetFile 에 별도 임시 디렉토리를 만들어야 하는 경우가 있습니다 . Portal Server 의 모든 모듈에 공통된 임시 디렉토리를 지정하면 디스크 공간이 금방 부족해질 수 있습니다 . 임시 디렉토리에 공간이 없으면 NetFile 이 작동하지 않습니다 .
파일 업로드 한계 (MB)	5	업로드할 수 있는 최대 파일 크기를 지정합니다 . 잘못된 값을 입력하면 NetFile 이 잘못된 값을 기본값으로 재설정합니다 . 정수 값만 입력해야 합니다 . 사용자마다 다른 파일 업로드 제한 크기를 지정할 수 있습니다 .

검색

표 B-13 은 NetFile 서비스 작업 검색 속성의 목록입니다 .

표 B-13 NetFile 서비스 작업 - 검색 속성

속성	기본값	설명
디렉토리 검색 제한	100	한 번의 검색으로 검색되는 최대 디렉토리 수를 지정합니다 .

압축

표 B-14 는 NetFile 서비스 작업 압축 속성의 목록입니다 .

표 B-14 NetFile 서비스 작업 - 압축 속성

속성	기본값	설명
기본 압축 유형	Zip	Zip 또는 Gzip 압축 유형을 지정합니다 .
기본 압축 수준	6	1 과 9 사이에서 압축 수준을 지정합니다 .

일반

표 B-15 는 NetFile 서비스 일반 속성의 목록입니다 .

표 B-15 NetFile 서비스 ñ 일반 속성

속성	기본값	설명
MIME 유형 구성 파일 위치	/opt/S1PS62/SUNWps/samples/config/netfile	클라이언트 브라우저로 보낼 응답 컨텐트 유형을 지정합니다 .

Netlet 서비스

표 B-16 은 Netlet 서비스 속성의 목록입니다 .

표 B-16 Netlet 서비스 속성

속성	기본값	설명
Netlet 규칙		규칙의 추가 또는 삭제를 선택합니다 .
규칙을 추가하면 다음 9 가지 속성이 필요합니다 .		
-- 규칙 이름		규칙에 대한 고유 이름을 지정합니다 .
-- 암호화 암호		필요한 암호를 지정합니다 .
--URL		호출될 응용프로그램에 대한 URL 을 지정합니다 .
-- 애플릿 다운로드		애플릿을 다운로드해야 하는지를 지정합니다 . 애플릿이 사용되는 경우 , 관련 편집 상자의 구문은 다음과 같습니다 . *local-port:server-host:server-port*
-- 세션 확장		이 규칙에 해당하는 Netlet 세션이 실행되는 동안에 Portal Server 세션 시간이 연장되도록 합니다 .

표 B-16 Netlet 서비스 속성

속성	기본값	설명
-- 로컬 포트를 대상 서버 포트에 매핑		로컬 포트, 대상 호스트 및 대상 포트를 지정합니다. 이 값을 입력한 후 (이 표의 다음 3 개 행에) [추가] 를 눌러 목록에 나타나도록 합니다.
-- 로컬 포트		Netlet 이 수신할 로컬 포트를 지정합니다. FTP 규칙의 경우 로컬 포트 값은 30021 이어야 합니다.
-- 대상 호스트		정적 규칙에는 Netlet 연결 대상 컴퓨터의 호스트 이름이 포함됩니다. 동적 규칙에는 단어 "TARGET" 이 포함됩니다.
-- 대상 포트		대상 호스트의 포트를 지정합니다.
기본 원시 VM 암호		Netlet 규칙에 대한 기본 암호를 지정합니다. 규칙의 일부로 암호를 포함하지 않은 기존 규칙을 사용할 때 유용합니다.
기본 Java 플러그인 암호		Netlet 규칙에 대한 기본 암호를 지정합니다. 규칙의 일부로 암호를 포함하지 않은 기존 규칙을 사용할 때 유용합니다.
기본 루프백 포트	58000	Netlet 을 통해 애플릿을 다운로드할 때 클라이언트에서 사용할 포트를 지정합니다. Netlet 규칙에서 기본값을 무시할 수 있습니다.
연결 재인증		Netlet 연결을 구성해야 할 때마다 사용자가 Netlet 비밀번호를 입력하도록 합니다.
연결에 대해 경고 팝업 표시	선택	사용자가 Netlet 에서 응용 프로그램 실행할 때, 그리고 침입자가 수신 포트를 통해 데스크탑에 액세스하려고 할 때 메시지가 표시됩니다.
포트 경고 대화 상자에 확인란 표시	선택	Netlet 에서 사용자의 표준 포털 데스크탑에 있는 대상 호스트에 연결하려 할 때 경고 대화 상자 팝업을 억제하는 옵션을 사용자에게 제공합니다.
연결 유지 시간 (분)	0	클라이언트에서 웹 프록시를 통해 게이트웨이에 연결하는 경우에는 유휴 Netlet 연결이 프록시 시간 초과로 해제됩니다. 이를 방지하려면 이 매개 변수에 프록시 시간 초과보다 작은 값을 입력합니다.
포털 로그아웃할 때 Netlet 종료	선택	사용자가 Portal Server 를 로그아웃할 때 모든 연결이 종료되도록 합니다.
Netlet 규칙에 액세스	*	특정 조직, 역할 또는 사용자의 특정 Netlet 규칙에 대한 액세스를 정의합니다.
Netlet 규칙 거부		특정 조직, 역할 또는 사용자의 특정 Netlet 규칙에 대한 액세스를 거부합니다.
허용된 호스트	*	특정 조직, 역할 또는 사용자의 특정 호스트에 대한 액세스를 정의합니다.

표 B-16 Netlet 서비스 속성

속성	기본값	설명
거부된 호스트		조직 내에서 특정 호스트에 대한 액세스를 거부합니다 .

Proxylet 서비스

표 B-17 은 Proxylet 서비스 속성을 나열합니다 .

표 B-17 Proxylet 서비스 속성

속성	기본값	설명
Proxylet 애플릿 자동으로 다운로드	127.0.0.1	이 확인란을 선택하면 , 사용자가 로그인할 때 클라이언트 컴퓨터에 Proxylet 이 다운로드됩니다 .
기본 Proxylet 애플릿 바인딩 IP	127.0.0.1	Proxylet 애플릿이 있는 IP 주소 .
기본 Proxylet 애플릿 포트	58080	Proxylet 이 청취하는 포트입니다 .

국가 코드

다음 표에는 인증서 관리 중에 지정해야 하는 2 자로 된 국가 코드가 나열되어 있습니다 .

표 C-1 2 자로 된 국가 코드 *(1 / 10)*

ad	안도라
ae	아랍에미리트
af	아프가니스탄
ag	앤티가 바부다
ai	앙귈라
al	알바니아
am	아르메니아
an	네덜란드령 안틸레스
ao	앙골라
aq	남극
ar	아르헨티나
arpa	구식 아르파네트
as	미국령 사모아
at	오스트리아
au	오스트레일리아
aw	아루바
az	아제르바이젠

표 C-1 2 자로 된 국가 코드 *(2 / 10)*

ba	보스니아 헤르체고비나
bb	바베이도스
bd	방글라데시
be	벨기에
bf	부르키나파소
bg	불가리아
bh	바레인
bi	부룬디
bj	베냉
bm	버뮤다
bn	브루나이
bo	볼리비아
br	브라질
bs	바하마
bt	부탄
bv	부베이 섬
bw	보츠와나
by	벨로루시
bz	벨리즈
ca	캐나다
cc	코코스 군도
cf	중앙 아프리카
cd	콩고 민주 공화국
cg	콩고
ch	스위스
ci	코트디부와르
ck	쿠크 군도

표 C-1 2 자로 된 국가 코드 *(3 / 10)*

cl	칠레
cm	카메룬
cn	중국
co	콜롬비아
com	상용
cr	코스타리카
cs	구 체코슬로바키아
cu	쿠바
cv	카보베르데
cx	크리스마스 섬
cy	사이프러스
cz	체코
de	독일
dj	지부티
dk	덴마크
dm	도미니카
do	도미니카 공화국
dz	알제리
ec	에쿠아도르
edu	교육적
ee	에스토니아
eg	이집트
eh	서사하라
er	에리트레아
es	스페인
et	에디오피아
fi	핀란드

표 C-1 2 자로 된 국가 코드 *(4 / 10)*

fj	피지
fk	포클랜드
fm	마이크로네시아
fo	페로 군도
fr	프랑스
fx	프랑스 (유럽 영토)
ga	가봉
gb	영국
gd	그레나다
ge	그루지야
gf	프랑스령 가이아나
gh	가나
gi	지브랄타
gl	그린랜드
gm	감비아
gn	기니
gov	미 정부
gp	과달루프 (프랑스령)
gq	적도 기니
gr	그리스
gs	사우스 조지아 및 사우스 샌드위치 군도
gt	과테말라
gu	괌 (미국령)
gw	기니비사우
gy	가이아나
hk	홍콩
hm	허드 섬 및 맥도널드 군도

표 C-1 2 자로 된 국가 코드 *(5 / 10)*

hn	온두라스
hr	크로아티아
ht	아이티
hu	헝가리
id	인도네시아
ie	아일랜드
il	이스라엘
in	인도
int	국제
io	영인도 제도
iq	이라크
ir	이란
is	아이슬란드
it	이탈리아
jm	자메이카
jo	요르단
jp	일본
ke	케냐
kg	키르기스스탄
kh	캄보디아
ki	키리바시
km	코모로
kn	세인트 크리스토퍼 네비스
kp	북한
kr	남한
kw	쿠웨이트
ky	카이만 군도

표 C-1 2 자로 된 국가 코드 *(6 / 10)*

kz	카자흐스탄
la	라오스
lb	레바논
lc	세인트 루시아
li	리히텐슈타인
lk	스리랑카
lr	라이베리아
ls	레소토
lt	리투아니아
lu	룩셈부르크
lv	라트비아
ly	리비아
ma	모로코
mc	모나코
md	몰다비아
mg	마다가스카르
mh	마셜 군도
mil	미군
mk	마케도니아
ml	말리
mm	미얀마
mn	몽골
mo	마카오
mp	북마리아나 군도
mq	말티니크 (프랑스령)
mr	모리타니
ms	몬트세라트

표 C-1 2 자로 된 국가 코드 *(7 / 10)*

mt	몰타
mu	모리셔스
mv	몰디브
mw	말라위
mx	멕시코
my	말레이시아
mz	모잠비크
na	나미비아
nato	NATO(이 페이지는 1996 년 삭제되었음 -hq.nato.int 참조)
nc	뉴 칼레도니아 (프랑스령)
ne	니제르
net	네트워크
nf	노퍽 섬
ng	나이지리아
ni	니카라과
nl	네덜란드
no	노르웨이
np	네팔
nr	나우루
nt	중립 지역
nu	니우에
nz	뉴질랜드
om	오멘
org	비영리 조직 (sic)
pa	파나마
pe	페루
pf	폴리네시아 (프랑스령)

표 C-1 2 자로 된 국가 코드 *(8 / 10)*

pg	파푸아뉴기니
ph	필리핀
pk	파키스탄
pl	폴란드
pm	세인트 피에르 미켈론
pn	핏케언 군도
pr	푸에르토리코
pt	포르투갈
pw	팔라우
py	파라과이
qa	카타르
re	리유니언 (프랑스령)
ro	루마니아
ru	러시아
rw	르완다
sa	사우디아라비아
sb	솔로몬 군도
sc	세이셸
sd	수단
se	스웨덴
sg	싱가포르
sh	세인트 헬레나
si	슬로베니아
sj	스발바르드 얀마이옌 군도
sk	슬로바키아
sl	시에라리온
sm	산마리노

표 C-1 2 자로 된 국가 코드 *(9 / 10)*

sn	세네갈
so	소말리아
sr	수리남
st	상투메 프린시페
su	구 USSR
sv	엘살바도르
sy	시리아
sz	스와질랜드
tc	터크스 카이코스 군도
td	차드
tf	프랑스 남부 지방
tg	토고
th	태국
tj	타지키스탄
tk	토켈라우
tm	투르크메니스탄
tn	튀니지
to	통가
tp	동티모르
tr	터키
tt	트리니다드 토바고
tv	투발루
tw	대만
tz	탄자니아
ua	우크라이나
ug	우간다
uk	영국

표 C-1 2 자로 된 국가 코드 *(10 / 10)*

um	미국령 군도
us	미국
uy	우루과이
uz	우즈베키스탄
va	바티칸 시국
vc	세인트 빈센트 그레나딘스
ve	베네수엘라
vg	버진 군도 (영국령)
vi	버진 군도 (미국령)
vn	베트남
vu	바누아투
wf	월리스 푸투나
ws	사모아
ye	예멘
yt	마요트
yu	유고슬라비아
za	남아프리카
zm	잠비아
zr	자이르
zw	짐바브웨

용어

이 설명서 세트에서 사용되는 용어의 전체 목록은 Java Enterprise System Glossary(http://docs.sun.com/doc/816-6873) 을 참조하십시오 .

색인

C

CCS(Cascading Style Sheet)
  Rewriter 에서 124
certadmin 스크립트 216
chroot 47
Citrix
  html 파일 203

D

DMZ 28
DNS 200

E

EProxy 185

G

gwmultiinstance 스크립트 50

H

HTML
  Rewriter 의 규칙 99

HTTP
기본 인증 246
리소스 , 연결 56
웹 프록시를 사용하는 리소스 56
헤더 73

I

iNotes 35

J

JavaScript
Rewriter 의 규칙 106

M

Messenger Express 35
Microsoft Exchange 서버 201
MIME
구문 분석할 유형 276
추측 131, 278
MIME 유형
목록 만들기 276
지정 304

N

NetFile 177
Unix 인증 180
공통 호스트 목록 288
구성 285
디버깅 305
로깅 180
소개 177
압축 303
액세스 사용 설정 179
업로드 제한 크기 301
임시 파일 디렉토리 300
지원되는 프로토콜 178
창 위치 299
창 크기 298
호스트에 액세스 293
호스트에 액세스 거부 295
호스트에 액세스 허용 294
NetFile 서비스 32
Netfile 에서 178
Netlet 184
PDC 를 위한 구성 207
Sun Ray 환경에서 203
공급자 185
구성 307
구성 요소 184
규칙 185, 187
규칙 추가 309
로그아웃할 때 종료 315
로깅 202, 284
사용 시나리오 186
소개 183
수신 포트 184
애플릿 184
애플릿 다운로드 확인란 309
연결 유지 시간 315
연결에 대한 재인증 활성화 313
원격 호스트에서 애플릿 다운로드 187
웹 프록시를 통한 터널링 263
포트 번호 194

호스트에 액세스 318
호스트에 액세스 거부 318
활성화 243
Netlet 규칙 310
동적 191
삭제 311
수정 310
액세스 거부 317
액세스 지정 316
정적 규칙 191
Netlet 규칙 예제
FTP 202
IMAP 200
Lotus Notes 비 웹 클라이언트 200
Lotus 웹 클라이언트 200
Microsoft Outlook 및 Exchange 서버 201
Netscape 4.7 우편 클라이언트 202
SMTP 200
Netlet 서비스 32
사용자에 지정 308
Netlet 프록시 185
다시 시작 69
만들기 68
사용 65
이점 65
활성화 69
nlpmultiinstance 스크립트 68
Novell Netware
protocol 179

O

OS 문자 집합 286
Outlook Web Access 201
구성 174
규칙 집합 174, 275

P

PAC
구성 62
PDC 269
구성 207, 269
인증 210
인증 체이닝 76
platform.conf 40
등록 정보 41
Portal Server
목록 만들기 254
ProFTPD
파일 업로드 179
Proxylet
구성 321
소개 88
이점 88
활성화 245
Proxylet 서비스 32

R

Rewriter
6.x 규칙 집합을 3.0 과 매핑 175
HTML 규칙 99
JavaScript 규칙 106
URLScraper 91
XML 규칙 121
구문 분석기와 URI 의 매핑 목록 만들기 131

구성 125
규칙 작성 97
규칙 집합 DTD 92
규칙 집합에 대한 URI 의 매핑 목록 만들기 273
규칙에서 패턴 매칭 105
다시 쓰지 않을 URI 목록 만들기 127
디버깅 로그 사용 135
마스크 활성화 132
모든 URL 의 다시 쓰기 사용 272
모든 URL 의 재작성 126
및 도메인 및 부속 도메인의 프록시 목록 61
사례 연구 170
예제 137
와일드카드 사용 127
일관된 프로토콜 282
작업 예제 137
Rewriter 서비스 31
Rewriter 프록시
다시 시작 72
만들기 70
이점 70
활성화 72
RProxy 185
rwpmultiinstance 70

S

Secure Sockets Layer 30
SMTP 243
SRA
모든 속성 목록 335
서비스 31
소프트웨어 27
접속 지원 56
SSL 210
SSL 버전 3.0 267
SSO 235

T

TCP/IP 184, 243

U

UNIX
명령줄 32
인증 180
URL
거부된 234
다시 쓰기 사용 272
다시 쓰지 않음 278
동적 Netlet 규칙으로 불러옴 197
사용자 세션 쿠키 250
세션 얻기 256
웹 프록시용 목록 만들기 258
인증되지 않은 264
허용된 235
URLScraper 91

W

Windows
도메인 291, 292
Windows 도메인
지정 291
WINS/DNS 서버
지정 292

WML
Rewriter 의 규칙 125

X

XML 규칙
Rewriter 에서 121

ㄱ

가속기
Sun Crypto 1000 323
Sun Crypto 4000 327
외부 SSL 장치 330
프록시 330
거부
URL 234
검색
디렉토리 제한 302
게이트웨이 54
chroot 모드 47
HTTP 모드 241
HTTPS 모드 241
URL 에서 세션 얻기 256
게이트웨이 프로필 38
구성 239
다중 인스턴스 50
로깅 283
소개 37
스레드 풀 지정 253
시간 초과 252
시작 53
여러 홈 52
연결 설정 241
인증서 사용 가능 264
중지 54
게이트웨이 서비스 31
게이트웨이 프로필
만들기 50
경고 팝업 대화 상자 313
공통 호스트 목록
구성 288
관리 콘솔 32
관리자 구성 암호 192
구문 분석기와 URI 의 매핑 131
구성
HTTP 지속 연결 사용 247
Netlet 307
Outlook Web Access 174
Proxylet 321
Rewriter 125
Secure Remote Access 33
개인 디지털 인증서 269
거부된 URL 234
거부된 호스트 목록 295
게이트웨이 239
공통 호스트 목록 288
허용된 호스트 목록 294
구성 요소
Netlet 184
국가 코드
2 자로 된 값 351
규칙
CCS(Cascading Style Sheet) 124
Netlet 187
Rewriter 97
Rewriter 의 HTML 99
Rewriter 의 JavaScript 106
WML 125

추가 309
규칙 집합
OWA 275
일반 127
규칙 집합 매핑
URI 목록 만들기 273
기본 도메인
다시 쓰기 62
지정 277, 290
기본 암호화 암호 311
기본값
Windows 도메인 291, 292
Windows 워크그룹 291
게이트웨이 프로필 38
도메인 62

ㄴ

네트 규칙
예제 199

ㄷ

다시 시작 54
Netlet 프록시 69
Rewriter 프록시 72
게이트웨이 54
다중 인스턴스
게이트웨이 50
단일 사인온 235
대상 포트 186
도메인 및 부속 도메인의 프록시 59
동적 규칙
Netlet 191
불러옴 197
애플릿 다운로드 198
등록 정보
platform.conf 41
디버깅
NetFile 305
정보의 위치 320
디버깅 로그
Rewriter 135

ㄹ

로그 파일
파일 이름 333
로깅
NetFile 180
Netlet 202
Rewriter 135
게이트웨이 283
디버깅 활성화 320
루프백 포트
할당 312

ㅁ

마스크
활성화 132, 280
마스킹
씨드 문자열 280
만들기
Portal Server 목록 254
Rewriter 프록시 70
게이트웨이 프로필 38, 50
구문 분석기와 URI 의 매핑 목록 131
구문 분석할 매핑 목록 279

규칙 집합에 대한 URI 의 목록 273
다시 쓰지 않을 URI 목록 127, 278
마스크하지 않을 URI 목록 281
신뢰할 수 있는 SSL 도메인 목록 268
인증되지 않은 URL 목록 264
인증서 사용 가능 게이트웨이 호스트 목록 264
호스트 프록시 56
명령줄 프롬프트 25
모드
HTTP 241
HTTPS 241
보안 29
열린 28
문제 해결 135

ㅂ

보안 모드 29
브라우저 캐싱
사용 해제 77
비무장 지대 28
비밀번호
프록시 261
비활성화
Netlet 프록시 244
SSL 버전 2.0 266
단일 사인온 236
브라우저 캐싱 77

ㅅ

사례 연구
Rewriter 170
사용자 구성 가능 암호 192
사용자 세션 쿠키 전달 250
사용자 정의
게이트웨이 사용자 인터페이스 78
삭제
Netlet 규칙 311
생성
직접 서명한 인증서 217
서버 재시도 간격 255
서비스
SRA 31
선택
암호 266
소켓
연결 시간 초과 249
속성
SRA 335
구성 33
수정
Netlet 규칙 310
시간 초과
게이트웨이 252
소켓 연결 249
유예 249
캐시된 소켓 253
시작
게이트웨이 53
신뢰할 수 있는 SSL 도메인 268
실행
HTTP 모드 241
HTTPS 모드 241
응용프로그램 183

ㅇ

알림 34

암호
null 268
관리자 구성 192
기본 암호화 311
사용자 구성 가능 192
선택 266
지원되는 193
압축
NetFile 용 303
애플릿 184
다운로드 198
액세스
호스트에 거부 318
호스트에 허용 318
액세스 목록
거부된 URL 234
단일 사인온 235
허용된 URL 235
업로드 제한 크기
NetFile 용 301
역 프록시 73
활성화 73
연결
경고 팝업 대화 상자 활성화 313
지속 247
연결 유지 시간
설정 315
연합 관리 79
열린 모드 28
예제
Rewriter 137
와일드카드
Rewriter 에서 127
웹 프록시에서 58
와일드카드 인증서 77
워치독
Netlet 프록시 69
Rewriter 프록시 72
웹 프록시 56
Netlet 터널링 263
활성화 258
유예 시간 초과 249
응용프로그램
실행 183
지원되는 35
인증
PDC 76, 210
UNIX 180
체이닝 76
인증 수준 237
인증서
CA 로부터 설치 222
certadmin 스크립트 216
SSL 210
공인 인증서 212
루트 CA 인증서 221
루트 CA 인증서 나열 227
모두 나열 229
삭제 224
와일드카드 77
인쇄 230
인증서 서명 요청 219
주문 222
직접 서명 217
트러스트 속성 211, 212
트러스트 속성 수정 226
파일 210
일관성 282
임시 파일 디렉토리 300

ㅈ

자동 검색 178
자동 프록시 구성
  파일 위치 262
재인증
  연결 313
정적 규칙 191
종료
  Netlet 315
중지
  게이트웨이 54
지원 암호 193
지정 237
  MIME 유형 파일 304
  NetFile 창 위치 299
  NetFile 창 지정 298
  OS 문자 집합 286
  Windows 도메인 291
  게이트웨이 스레드 풀 크기 253
  게이트웨이 시간 초과 252
  기본 WINS/DNS 292
  기본 도메인 277, 290
  디렉토리 검색 제한 302
  루프백 포트 312
  연결 유지 시간 315
  인증 수준 237
  임시 디렉토리 300
  직접 연결 259
  최대 연결 대기 길이 251
  충돌 해결 34
  캐시된 소켓 시간 초과 253
  프록시 258
  프록시 인증 261
  호스트에 액세스 293
직접 서명한 인증서 217

ㅊ

창 위치
  NetFile 용 299
창 크기
  NetFile 용 298
처리 순서
  프록시 59
최대 요청 수 248
충돌 해결 34

ㅋ

캐시된 소켓 시간 초과 253
캘린더 35
쿠키
  안전하다고 표시 257
  외부 저장 255
  활성화 245

ㅌ

타사
  웹 사이트 참조 26
텔넷 243
트러스트 속성 211

ㅍ

파일 업로드 제한 301
포트
  Netlet 184
  대상 186

루프백 312
포트 번호
Netlet 194
표시 제거
포트 경고 314
프로토콜
NetFile 178
프록시
EProxy 185
Netlet 185, 244
Rewriter 241
RProxy 185
가속기 330
비밀번호 261
역 73
웹 56
인증 261
지정 258
호스트 프록시 지정 56
프록시 자동 구성 62
프롬프트
명령줄 25

ㅎ

할당
Netlet 서비스를 사용자에 308
기본 루프백 포트 312
허용
40 비트 브라우저 연결 265
허용된 URL 235
헤더
HTTP 73
호스트
액세스 거부 318
액세스 지정 293
액세스 허용 318
허용된 호스트 목록 구성 294, 295
호스트 프록시
만들기 56
홈이 여러 개인 게이트웨이 52
확인란
포트 경고 대화 상자에서 활성화 314
활성화
40 비트 브라우저 연결 265
HTTP 기본 인증 246
HTTP 연결 247
MIME 추측 131, 278
NetFile 액세스 179
Netlet 기록 202, 284
Netlet 프록시 69, 244
null 암호 268
PDC 인증 269
Proxylet 245
Rewriter 프록시 72, 241
SSL 버전 2.0 266
SSL 버전 3.0 267
경고 팝업 대화 상자 313
단일 사인온 236
디버깅 305
로깅 283
마스크 132, 280
모든 URL 의 재작성 126
암호 선택 266
역 프록시 73
연결 241
연결에 대한 재인증 313
외부 서버 쿠키 저장 255
웹 프록시 258
웹 프록시 사용 258
인증 체이닝 76

쿠키 245
쿠키를 안전하다고 표시 257
포트 경고 대화 상자에 확인란 표시 314