Interactive Session Recorder

Security Guide

Release 6.1

F20201-01

January 2020

ORACLE

Interactive Session Recorder Security Guide, Release 6.1

F20201-01

Contents

About This Guide

1 ISR Security Overview

About This Guide

The Interactive Session Recorder (ISR) Security Guide provides information about security considerations and best practices from a network and application security perspective for the ISR product.

Related Documentation

The following table describes the documentation set for this release.

Document Name	Document Description
ISR Release Notes	Contains information about new ISR features, fixes, and known issues.
ISR Installation Guide	Provides an overview of the ISR, hardware/software requirements and recommendations, storage considerations, pre-installation information, installation procedures, post-install verification procedures, making the first call, and additional advanced topics about the ISR.
ISR User Guide	Contains information about using the ISR Dashboard for all levels of users. Provides information about viewing, playing, deleting recordings, running reports, and managing user profiles.
ISR Administrator Guide	Contains information about using the ISR Dashboard for the Administrator level user (Super User, Account Administrator, Tenant Administrator). Provides information about creating and managing accounts, routes, and users. Also provides information about configuring the ISR, running reports, viewing active calls, and securing the ISR deployment.
ISR API Reference Guide	Contains information about ISR FACE, Recording File Types/Formats Supported, Return Codes, and Troubleshooting.
ISR Monitoring Guide	Contains information about installing and configuring the ISR Monitor, the Monitor database schema, and the Monitor MIB.
ISR Security Guide	Contains information about security considerations and best practices from a network and application security perspective for the ISR product.

Revision History

Date	Description
January 2018	• Initial release of ISR 6.1 software.
August 2018	• Adds "ISR Dashboard Cookies".
January 2020	• Adds a note regarding CA-signed certificates in "Signing Keys".

1

ISR Security Overview

This chapter describes how to configure security on the ISR.

Secure Installation

Security begins during ISR installation and choosing appropriate settings during installation helps protect your systems and data. Ensure that the critical security services and settings (described below) are installed and enabled. Oracle strongly recommends using a non-root account for logins to setup, configure, and administer your ISR systems. Choose secure passwords during installation and do not remove secure file permissions settings unless absolutely necessary.

Critical Security Services and Settings

By default, Oracle Linux 7 comes wiht several security features enabled. To help ensure the security of your systems, Oracle recommends that you do not disable these features.

- Firewalld—On Oracle Enterprise Linux 7, the firewalld services replaces the configuration elements of iptables from previous versions of Enterprise Linux. Keeping the firewalld service enabled and active provides an excellent defensive measure to secure your systems. For more information on the firewalld service, see http://docs.oracle.com/cd/E52668_01/E54669/E54669.pdf, section 26.3. By default, the ISR platform utilizes the zones detailed below, and our applications install firewalld service configurations to enable standard communications amongst the various zones. To change the zones on which an application is allowed to operate, see the section “Firewalld Optional Configuration” in this guide.
- SELinux/seten force—Provides an enhanced level of control over the files, processes, and users of the Operating System. For more information on the SELinux/seten force, see http://docs.oracle.com/cd/E52668_01/E54669/E54669.pdf, section 26.2.

Creating and Using a Non-Root User Account

Oracle strongly recommends using a non-root account for logins to setup, configure, and administer your ISR systems. Instead, create a normal user account in the 'isr' group.
To create a new user in the 'isr' group:

1. Add the new user by executing the following command:

   ```
   [root@localhost ~]# useradd -g 9001 <username>
   ```

2. Set the user's password by executing the following command:

   ```
   [root@localhost ~]# passwd <password>
   ```

3. Grant the user sudo permissions by adding them to the wheel group:

   ```
   [root@localhost ~]# usermod -aG wheel <username>
   ```

4. Verify you can use the new user account and the sudo permissions are configured correctly.

   ```
   # logout
   Localhost login: isradm
   ```

```
Password: **********
[isradm@localhost ~]$ touch /var/log/messages
touch: cannot touch ‘/var/log/messages’: Permission denied
[isradm@localhost ~]$ sudo touch /var/log/messages
[isradm@localhost ~]$
```

File Permissions

Do not unnecessarily remove file permission restrictions on files and directories. By default, ISR files are set to the most restrictive possible settings required for the system to operate.

Secure Passwords

Oracle recommends you use unique and complex passwords for ISR database accounts, as well as OS user accounts. The following Oracle MySQL password rules offer a good starting point:

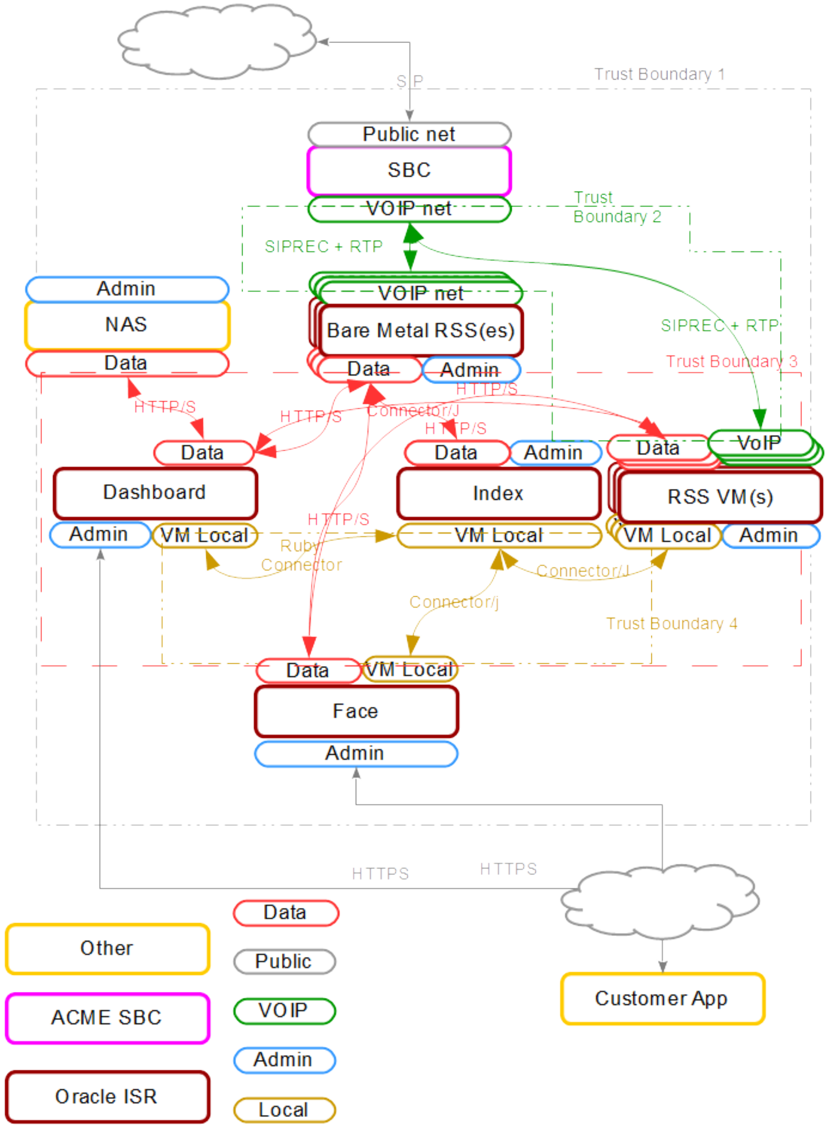
- At least 8 characters long
- Contain at least 1 uppercase and 1 lowercase letter
- Contain at least 1 number
- Contain at least 1 special character

Firewalld Configuration Overview

The firewalld service provides a strong line of defense in securing ISR Servers and Services. The firewall is, by default, enabled and configured to provide a secure operating environment for ISR. There are three default zones utilized by ISR services:

- Public—The default firewall zone interfacing to the most networks; This zone is utilized by the ‘Admin’ Ethernet interface. Services utilizing this zone include:
  - SSH
  - ISR Dashboard (HTTPS)
- Trusted—An internal firewall zone used by Data services such as:
  - MySQL (for non-VM RSS hosts)
  - ISR Web Services (HTTPS)
  - ISR Web Services (HTTP)
  - VoIP traffic (SIPREC/RTP)
- Internal—An internal firewall zone used by ISR VMs for communication. Services include:
  - MySQL

ISR 6.0 Architecture Reference



ISR Firewalld Configuration

By default, ISR provides a secure default firewall configuration which should not require end user changes. However, it may be necessary to modify these settings to disable unnecessary ISR services, or to allow communication with third party services. To help ensure the security of your systems, it is recommended that you do not disable the firewall.

- Service Configuration Files—ISR provides firewall zone configuration files, found in the `/opt/isr/security/firewalld/services/` directory. These files outline the services and ports utilized by the particular ISR service and configure the firewalld service to allow these communications.
- Interface/Zone Settings—ISR configures the firewall based on the “ISR Network Interface Mapping” performed during initial configuration.

- Service/Zone Settings—ISR comes preconfigured to allow the ISR Services to be run only on specified zones.

Modifying ISR Firewalld Configuration

By default, the firewall is configured upon installation to allow all services to communicate on specified interfaces within the firewalld zones. However, you may need to move a service to an additional zone, or remove an extraneous firewall service from a particular zone.

Common changes include:

- Adding the ISR Dashboard service to the public zone if it must be reachable from external addresses. This can be done by entering the following commands on the ISR Dashboard host:

  ```
  $ sudo firewall-cmd --zone=public --add-service dashboard
  $ sudo firewall-cmd --zone=public --add-service dashboard --permanent
  ```

  Similarly, it can be removed from the internal zone:

  ```
  $ sudo firewall-cmd --zone=internal --remove-service dashboard
  $ sudo firewall-cmd --zone=internal --remove-service dashboard --permanent
  ```

- Disabling unused components such as the ISR converter service.

  ```
  $ sudo firewall-cmd --zone=data --remove-service converter
  $ sudo firewall-cmd --zone=data --remove-service converter --permanent
  ```

ISR Port Usage

The ISR Platform utilizes the following ports, which are available on the networks displayed in the last column for each component host shown in the following table:

Component	Port	Description	Notes	Networks
All ISR Component Hosts	123	NTP		Admin
RSS	22*	SSH	SSL	Admin
	5060	SIP Listen Port (Recorder)		VoIP
	8080	HTTP Webserver		Data
	8443	Secure HTTP Webserver	SSL	Data
	9998	REST API Listen Port (Recorder)	SSL	Data
	9999	REST API Listen Port (Converter)	SSL	Data
	22000-46000	RTP		VoIP
Index	22*	SSH	SSL	Admin
	3306	MySQL		Local, Data
Dashboard	22*	SSH	SSL	Admin
	80	HTTP Webserver	Disabled /optional	Admin/External
	443	Secure HTTP Webserver	SSL	Admin/External
FACE	22*	SSH	SSL	Admin

Component	Port	Description	Notes	Networks
	8080	Web Service Port	Disabled /optional	Data
	8443	Web Service Port	SSL	Data

Note:

The ISR does not use port 22 within its system, however, it is typically open in the firewall for administrative connectivity.

ISR Certificates

Many ISR services are configured for more secure requests via HTTPS, including:

- ISR Dashboard
- ISR FACE
- Recorder REST Webservice
- Converter REST Webservice
- RSS Java API

To access these services, the clients you use must have either public keys or certificates, which are generated at installation time, or negotiated through a public key exchange. Public keys and certificates can be found in the locations described below.

The following table lists and describes the RSS public key locations.

Public Key Location	Description	Key Technology
`/opt/isr/security/keys/rss_cert.pem`	Certificate for ISR components to connect to RSS REST services	OpenSSL SHA256 RSA Key/ X509 Self-signed certificate
`/opt/isr/security/keys/isr.key`	Private key for ISR component communications	N/A
`/opt/isr/security/keys/israpi-public.key`	Public certificate for ISR API	Java keytool created RSA Key/ Certificate
`/opt/isr/security/keys/tomcat.keystore`	Keystore for ISR Java applications on the RSS	N/A

The following table lists and describes the Dashboard public key locations.

Public Key Location	Description	Key Technology
`/opt/isr/security/keys/puma.crt`	Certificate file	OpenSSL DES3 RSA Key/ X509 Self-signed certificate
`/opt/isr/security/keys/isr.key`	Private key for ISR component communications	N/A

The following table lists and describes the FACE public key locations.

Public Key Location	Description	Key Technology
/opt/isr/security/keys/face-public.key	Public key for FACE HTTPS clients	Java keytool created RSA Key/ Certificate
/opt/isr/security/keys/tomcat.keystore	Keystore for ISR Java applications on FACE	N/A
/opt/isr/security/keys/isr.key	Private key for ISR component communications	N/A

Imported Certificates for Secure Communications

Some ISRISRapplications (for example, the Dashboard) may send client requests to other ISR applications. For these requests and responses to be secure and authenticated, the application hosts must initially import the public keys of the services receiving the requests. The following table describes the keys imported to ISR component hosts for secure ISR application communication.

Component	Public Key Location	Description
Dashboard	/opt/isr/security/keys/israpi-public.key.<RSS host IP> /opt/isr/security/keys/rss_cert.pem.<RSS host IP>	Imported RSS API public key for Dashboard RSS API requests Imported RSS Converter and Recorder process public keys
FACE	opt/isr/security/keys/israpi-public.key.<RSS host IP> /opt/isr/security/keys/rss_cert.pem.<RSS host IP>	Imported RSS API public key for FACE RSS API requests Imported RSS Converter and Recorder process public keys

Signing Keys

Many ISR services utilize self-signed keys which are generated during installation. For better security, Oracle recommends that keys are signed by a Certificate Authority (CA). You must generate a certificate signing request (CSR) and use it to request a signed certificate from a CA. The certificates described in "Imported Certificates for Secure Communications" are self-signed when you install them. You must replace these with certificates signed by a certified Certificate Signing Authority (CSA).To obtain these properly signed certificates, you must generate a Certificate Signing Request (CSR).

To generate a CSR for your host:

1. Run /opt/isr/configIsr.sh from the Linux command line.
2. Choose the **'k' Manage ISR Keys** option.
3. Choose the **'c' Create Certificate Signing Request(s)** option.
4. Follow the instructions for creating a CSR.

CSRs are created in the `/opt/isr/security/keys/` directory.

Once you have generated a CSR, you must send it to a CSA for signing and install and replace the temporary self-signed certificate created during installation.

To import a signed certificate to your host:

1. Run /opt/isr/configIsr.sh from the Linux command line.
2. Choose the **'k' Manage ISR Keys** option.
3. Choose the **'i' Import a signed certificate** option.
4. Follow the instructions for importing your CA signed certificate.

> **Note:**
>
> If a CA-signed ISR API Face certificate has not been received, in bundled form, by the CA authority, then each signed certificate issued by the CA (for example, root certificates, intermediate certificates, and issued API Face signed certificates) must be manually imported using the below commands.
> The following command imports received root certificates to the tomcat keystore:
>
> ```
> keytool -import -file root.cert -alias root -keystore /opt/isr/security/
> keys/tomcat.keystore
> ```
>
> The following command imports received intermediate certificates to the tomcat keystore:
>
> ```
> keytool -import -file intermediate1.cert -alias intermed1 -
> keystore /opt/isr/security/keys/tomcat.keystore
> ```
>
> The following command imports received ISRAPI/Face certificates to the tomcat keystore:
>
> ```
> keytool -import -file CASigned_ISRAPI.cert -alias israpi-key -
> keystore /opt/isr/security/keys/tomcat.keystore
> ```
>
> Or:
>
> ```
> keytool -import -file CASigned_Face.cert -alias face-key -
> keystore /opt/isr/security/keys/tomcat.keystore
> ```

Additional CSR Details

You may need to attach additional information to your CSR. The following shows the general format for using keytool to create a CSR:

```
keytool -certreq -alias <alias> -keyalg RSA -file <alias>.csr -keystore /opt/isr/
security/keys/tomcat.keystore
```

The following shows the general format for using openssl to create a CSR:

```
openssl req -out <alias>.csr -key /opt/isr/security/keys/<keyfile> -new
```

Examples of Generating ISR Component CSRs

This section provides examples of generating ISR component CSRs.

RSS Certificate Signing

- RSS Services Certificate

  ```
  openssl req -out rss.csr -key /opt/isr/security/keys/rss_key.pem -new
  ```

- ISR API Certificate

  ```
  keytool -certreq -alias israpi-key -keyalg RSA -file israpi.csr -
  keystore /opt/isr/security/keys/tomcat.keystore
  ```

Dashboard Certificate Signing

- Dashboard Certificate

  ```
  openssl req -out dash.csr -key /opt/isr/security/keys/server.key -new
  ```

FACE Certificate Signing

- FACE API Certificate

  ```
  keytool -certreq -alias face-key -keyalg RSA -file face.csr -
  keystore /opt/isr/security/keys/tomcat.keystore
  ```

Managing Expired Keys and Certificates

Self-signed and CA-signed certificates both follow secure rules for expiration, and the expiration of these public keys and certificates impact ISR functionality if left unchecked.

Checking the Expiration Dates of ISR Keys

The following sections describe how to check the expiration dates of current certificates on each component host.

RSS

To check the expiration dates for RSS certificates, on the RSS host, execute the following commands and note the expiration dates in the output.

- RSS Java API

  ```
  $ keytool -v -printcert -file /opt/isr/security/keys/israpi-public.key |
  grep Valid
  ```

- Recorder and Converter webservices

  ```
  $ keytool -v -list -keystore /opt/isr/security/keys/tomcat.keystore | grep -
  A 8 israpi-key | grep Valid
  ```

Dashboard

To check the expiration dates for Dashboard certificates, on the Dashboard host, execute the following command and note the expiration date in the output.

```
$ keytool -v -printcert -file /opt/isr/security/keys/puma.crt | grep Valid
```

FACE

To check the expiration dates for FACE certificates, on the FACE host, execute the following commands and note the expiration dates in the output.

```
$ keytool -v -list -keystore /opt/isr/security/keys/tomcat.keystore | grep -A 8
face-key | grep Valid
```

Updating Expiring Self-Signed Keys

The following sections describe how to update expiring self-signed keys on each component host.

Expiring RSS Certificate

The following instructions describe how to update an expiring RSS certificate.

1. On the RSS host, move keys to an archive directory.

   ```
   $ mkdir /opt/isr/security/keys/old
   $ mv /opt/isr/security/keys/rss_*.pem /opt/isr/security/keys/old
   $ mv /opt/isr/security/keys/*public.key* /opt/isr/security/keys/old
   ```

2. Run the configIsr.sh script to regenerate the keys.

   ```
   $ sudo /opt/isr/configIsr.sh
   ```

   - Hit <Enter> and choose **yes** at the following prompt:

     ```
     Now Generating RSS key and certificate files. If you have not already
     configured the RSS data network IP address, please skip this key
     generation, configure networking and run the configuration option 'm'
     again.
     Hit <Enter> when ready.
     Continue generating key and certificate files: [yes]
     ```

   - Follow the configIsr script prompts closely.

3. On the Dashboard host, import the new keys.

   ```
   $ sudo /opt/isr/configIsr.sh
   ```

   - Choose the 'k' option to "Manage ISR keys".
   - Choose the 'r' option to "Import keys from an RSS".
   - Follow the script's instructions closely.

4. On the FACE host, import the new keys.

   ```
   $ sudo /opt/isr/configIsr.sh
   ```

   - Choose the 'k' option to "Manage ISR keys".
   - Choose the 'r' option to "Import keys from an RSS".
   - Follow the script's instructions closely.
   - Allow the export of the FACE key to the RSS to fail.

Expiring Dashboard Certificate

The following instructions describe how to update an expiring Dashboard certificate.

1. On the Dashboard host, move keys to an archive directory.

   ```
   $ mkdir /opt/isr/security/keys/old
   $ mv /opt/isr/security/keys/server.* /opt/isr/security/keys/old/
   $ mv /opt/isr/security/keys/puma.crt /opt/isr/security/keys/old/
   ```

2. Run the configIsr.sh script to regenerate the keys.

   ```
   $ sudo /opt/isr/configIsr.sh
   ```

   - Hit <Enter> and choose **yes** at the following prompt:

     ```
     Generating Private Key. Please enter a new key password when prompted.
     Please do not lose this password as it will be required throughout the
     installation process.
     Hit <Enter> when ready.
     Continue generating key and certificate files: [yes]
     ```

   - Follow the configIsr script prompts closely.

Expiring FACE Certificate

The following instructions describe how to update an expiring FACE certificate.

1. On the FACE host, move keys to an archive directory.

   ```
   $ mkdir /opt/isr/security/keys/old
   $ mv /opt/isr/security/keys/*.* /opt/isr/security/keys/old/
   ```

2. Run the configIsr.sh script to regenerate the keys.

   ```
   $ sudo /opt/isr/configIsr.sh
   ```

   - Hit <Enter> and choose **yes** at the following prompt:

     ```
     Now Generating RSS key and certificate files. If you have not already
     configured the RSS data network IP address, please skip this key
     generation, configure networking and run the configuration option 'm'
     again.
     Hit <Enter> when ready.
     Continue generating key and certificate files: [yes]
     ```

   - Follow the configIsr script prompts closely.

3. On the RSS host(s), remove the FACE public key from the keystore. Execute the following command and note the alias name.

   ```
   $ keytool -v -list -keystore /opt/isr/security/keys/tomcat.keystore | grep
   face
   $ sudo keytool -delete -alias face-key-<e.g. 10.10.20.30> -keystore /opt/isr/
   security/keys/tomcat.keystore
   ```

4. On the FACE host, import the RSS keys and export the new FACE key.

   ```
   $ sudo /opt/isr/configIsr.sh
   ```

   - Choose the 'k' option to "Manage ISR keys".
   - Choose the 'r' option to "Import keys from an RSS".
   - Follow the script's instructions closely.

5. Copy the original private key back into the keys directory.

   ```
   $ sudo cp /opt/isr/security/keys/old/isr.key /opt/isr/security/keys/
   ```

Configuring Reduced Security

The ISR's FACE functionality and Dashboard may all be run with reduced security. This section describes how to use the configCis.sh script to loosen security on these components.

Configuring FACE Reduced Security

The ISR's FACE functionality may be run with reduced security. You can use the configCis.sh script to loosen security settings on the FACE host.

1. To disable HTTPS in FACE, run the configCis.sh script and select HTTP for FACE.

```
[root@face ~]# configCis.sh
--------------------------------------------
Please select from the following menu:
--------------------------------------------

s) Show the current configuration
m) Modify the current configuration
i) Add/modify a second network interface
f) Set face default configuration in DB
q) Quit

Choice: f

WARNING, this action will reset the FACE to its default configuration.
  ** All customization of FACE or EEN configured will be lost.

Continue? (yes|no) [yes] yes
You have been warned.

Enter Face Host IP: [] 1.2.3.4
Protocol to use for FACE connections? (http|https) [https] http

FACE connection protocol set to http
Enter ObserveIT Server IP: [] 2.3.4.5
Protocol to use for ObserveIT Server connections? (http|https) [https]
ObserveIT connection protocol set to https
Attempting to restore backup SQL
Backing up FACE Config (to /opt/isr/faceSetupTemplate.sql.bak).
Updating FACE IP in SQL Script.
Updating FACE HTTP/S in SQL Script.
Updating ObserveIT IP in SQL Script.
```

ISR Dashboard Cookies

Dashboard cookies are set by default with a domain attribute of the empty string with the path attribute set to /. This results in a "host-only" cookie, with no subdomains included. ISR Dashboard administrators may need to manage these domain and path settings for security or functional purposes.

To configure Dashboard cookie attributes for specific domains and paths, create a backup file, and then edit the following file:

```
/var/www/dashboard/current/config/initializers/session_store.rb
```

Include the domain and path attributes after the "key" entry, separated by a comma.