IdP Basics & Installation

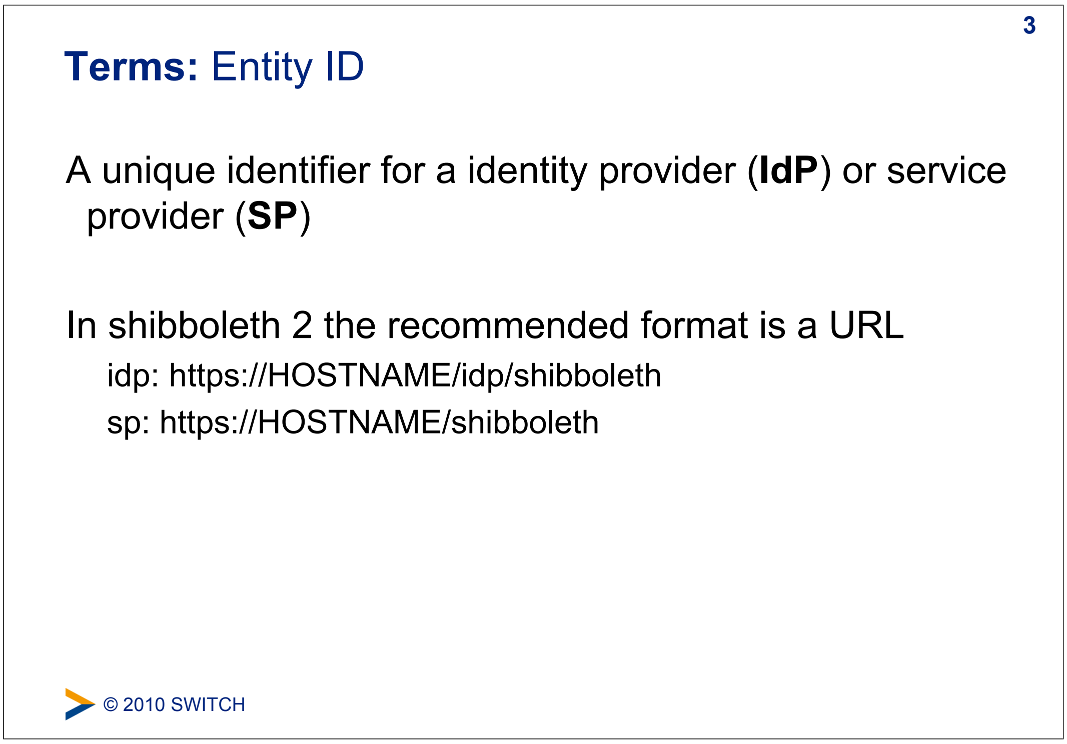
SWITCH
Serving Swiss Universities

Notes:

Current Environment

- Network
- Java
- Tomcat
- LDAP
  - Create apacheDS run directory
    ```
    mkdir /var/run/apacheds/default
    ```
  - Change permission to world writeable
    ```
    chmod a+w /var/run/apacheds/default
    ```

Notes:



3

Terms: Entity ID

A unique identifier for a identity provider (**IdP**) or service provider (**SP**)

In shibboleth 2 the recommended format is a URL

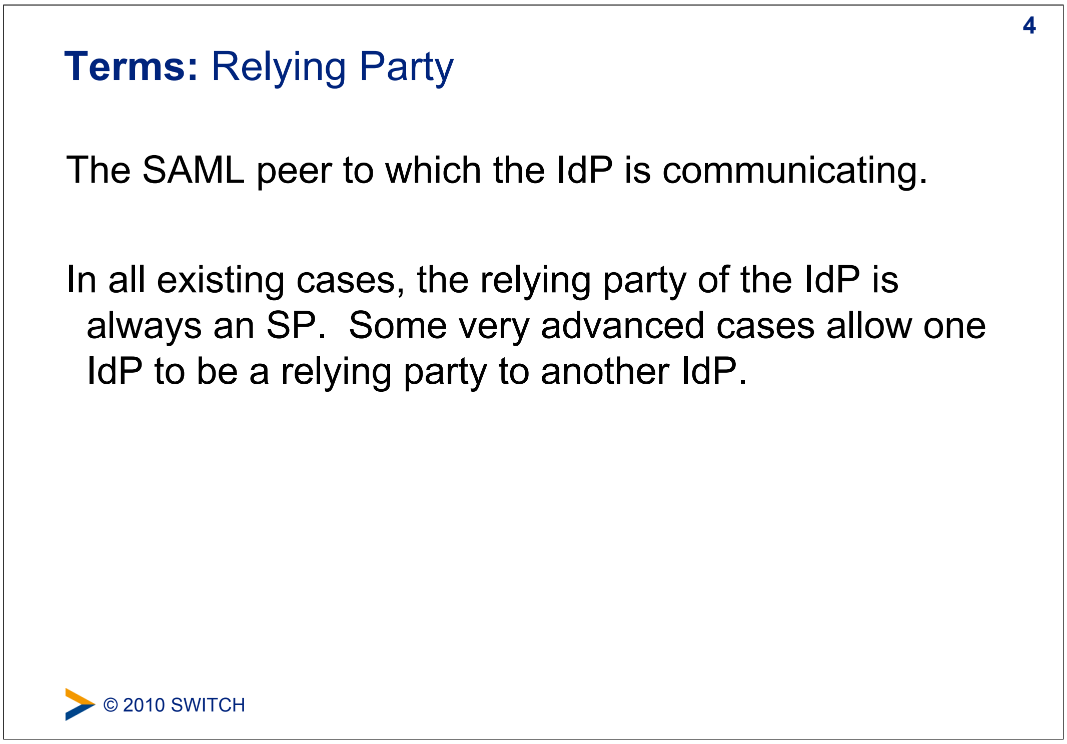
- idp: https://HOSTNAME/idp/shibboleth
- sp: https://HOSTNAME/shibboleth

Notes: ____

____

____

____



4

Terms: Relying Party

The SAML peer to which the IdP is communicating.

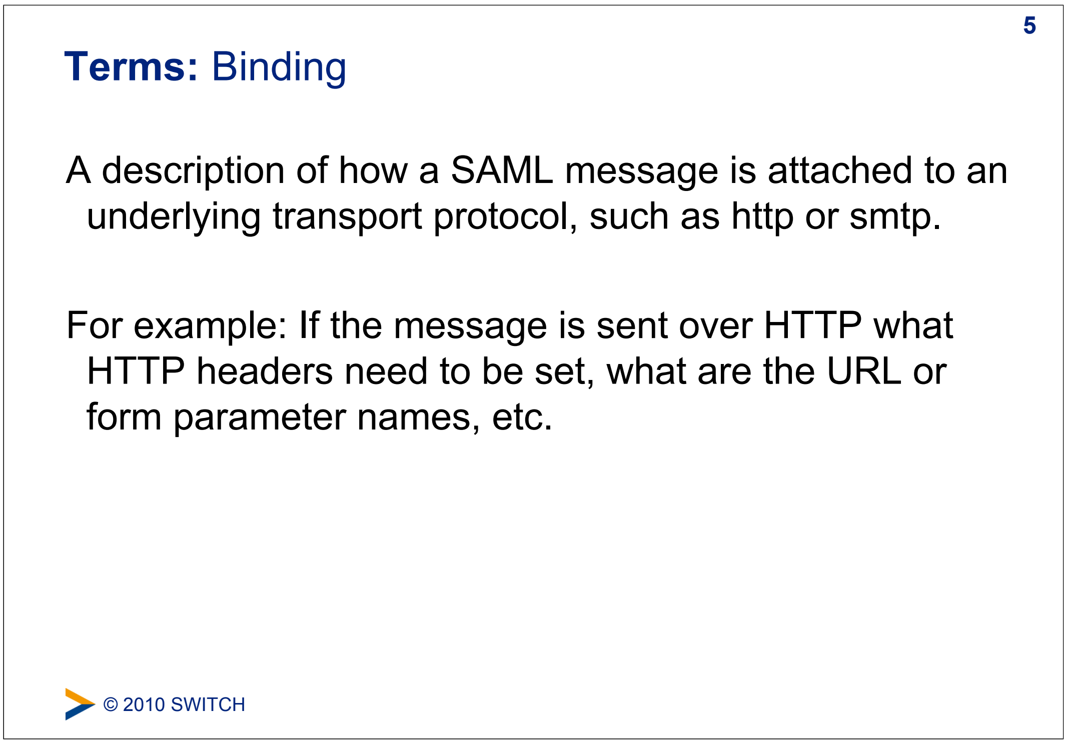
In all existing cases, the relying party of the IdP is always an SP. Some very advanced cases allow one IdP to be a relying party to another IdP.

Notes: ____

____

____

____



Terms: Binding

A description of how a SAML message is attached to an underlying transport protocol, such as http or smtp.

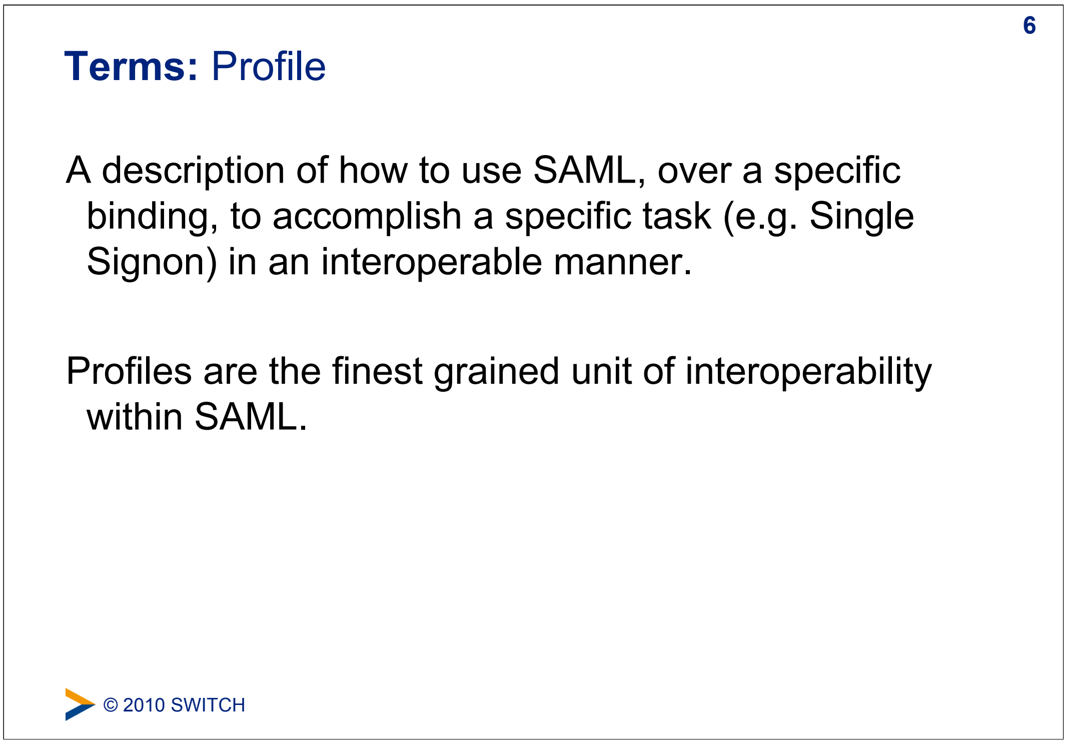
For example: If the message is sent over HTTP what HTTP headers need to be set, what are the URL or form parameter names, etc.

Notes: ____

____

____

____



Terms: Profile

A description of how to use SAML, over a specific binding, to accomplish a specific task (e.g. Single Signon) in an interoperable manner.

Profiles are the finest grained unit of interoperability within SAML.

Notes: ____

____

____

____

Terms: Metadata

A description of the SAML features supported by a SAML entity. Most importantly this includes the URLs for communicating with an entity.

Shibboleth also uses this information to build technical trust between entities.

Notes:

Installation

1. unzip /opt/installfest/distro/shibboleth-identityprovider-2.*-bin.zip
2. cd shibboleth-identityprovider-2.*
3. chmod +x install.sh
4. ./install.sh
   a. use **/opt/shibboleth-idp** as your shib home directory
   b. enter your hostname: **idp#.example.org**
   c. enter **password** for your password
5. cp -r endorsed /opt/tomcat/

Notes:

Tomcat Deployment

1. Create the file
   /opt/tomcat/conf/Catalina/localhost/idp.xml
   with the contents:

```
<Context
     docBase=”/opt/shibboleth-idp/war/idp.war”
     privileged="true”
     antiResourceLocking="false”
     antiJARLocking="false”
     unpackWAR="false" />
```

2. Start Tomcat with `$ tomcatStart`
3. Test your install

   https://idp#.example.org/idp/profile/Status

Notes:

SHIB_HOME

- /opt/shibboleth-idp should now contain:
  - bin
  - conf
  - credentials
  - lib
  - logs
  - metadata
  - war
- The Shibboleth documentation refers to this directory as SHIB_HOME

Notes:

SHIB_HOME/bin

Contains command line tools

aacli: Attribute authority command line interface allows you to simulate an attribute query/release

version: Provides the version of the IdP

Notes:

SHIB_HOME/conf

The IdP’s configuration files.

We’ll cover most as we go through the course. We will not cover service.xml or internal.xml as these control advanced features.

Notes:

SHIB_HOME/credentials

Credentials used by the IdP.

By default the IdP’s generated key (idp.key), cert (idp.crt) and a keystore (idp.jks) containing both are put here.

Good location to place things like trust anchor X.509 certs, cached CRLs, etc.

Notes: ____________________

____

____

____

SHIB_HOME/lib

The libraries (jars) that make up the IdP.

These are copies of those that occur in the IdP WAR file and are only used by the command line tools.

Notes: ____________________

____

____

____

SHIB_HOME/logs

Location of the Shibboleth log files.

process log: detailed description the IdP processing requests

access log: record of all the clients that connect to the idP

audit log: record of all information sent out from the IdP

Notes:

SHIB_HOME/metadata

Default location where various metadata files are stored.

The IdP does not automatically load any metadata. Metadata read from a file, or stored backup copies of remote metadata are usually put in this directory.

Notes:

SHIB_HOME/war

The location of the IdP WAR file created by the installer.

We point Tomcat to this file, instead of copying it to Tomcat, so that we don't forget to copy new WARs if we rebuild the IdP (to add an extension, for example) or run into problems with Tomcat's file caching mechanisms.

Notes:

Now some sleight of hand

```
cp /opt/installfest/idps/idp#/idp.*
  /opt/shibboleth-idp/credentials
```

```
cp /opt/installfest/sps/sp#/*.xml
  /opt/shibboleth-idp/metadata
```

Notes:

Logging: Configuration

- Logging configuration is controlled by the logging.xml config file
- Log messages belong in a hierarchal category; most correspond to Java package names
- Log messages have 5 levels: TRACE, DEBUG, INFO, WARN, ERROR

https://spaces.internet2.edu/display/SHIB2/IdPLogging

Notes:

Logging: Configuration

- Look at the IdP process log. Note how the messages are all info messages.
- Edit logging.xml and the change the logging level for the logger `edu.internet2.middleware.shibboleth` to DEBUG
- Restart the IdP and look at the process log again.
- The IdP will pick up change to logging.xml every 5 minutes.

Notes:

Metadata: Goals

- Load metadata for local SPs from the filesystem
- Load metadata for the “class” SPs from a remote location

Notes: ______________________________

____

____

____

Metadata: Configuration

- Metadata is loaded in to the IdP by **metadata providers**.
- Metadata providers are configured in the relying-party.xml file
- This file may only contain one top-level provider. By default the top level provider is a chaining provider that contains other metadata providers and uses them in the order defined.

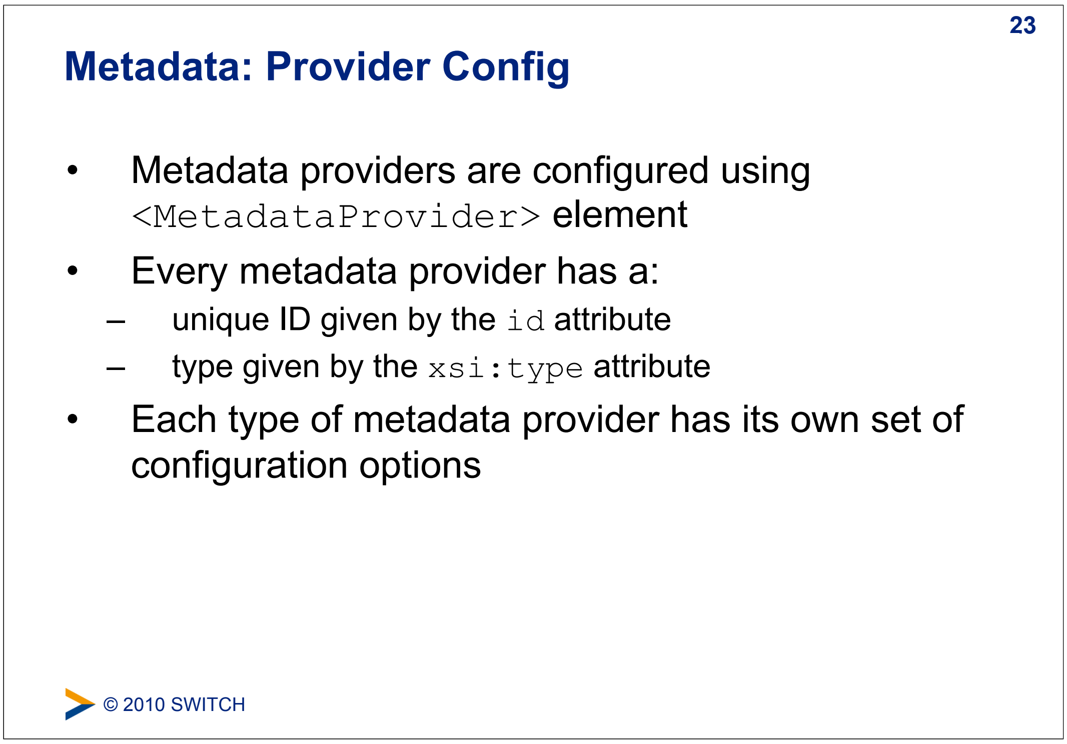
https://spaces.internet2.edu/display/SHIB2/IdPMetadataProvider

Notes: ______________________________

____

____

____



Metadata: Provider Config

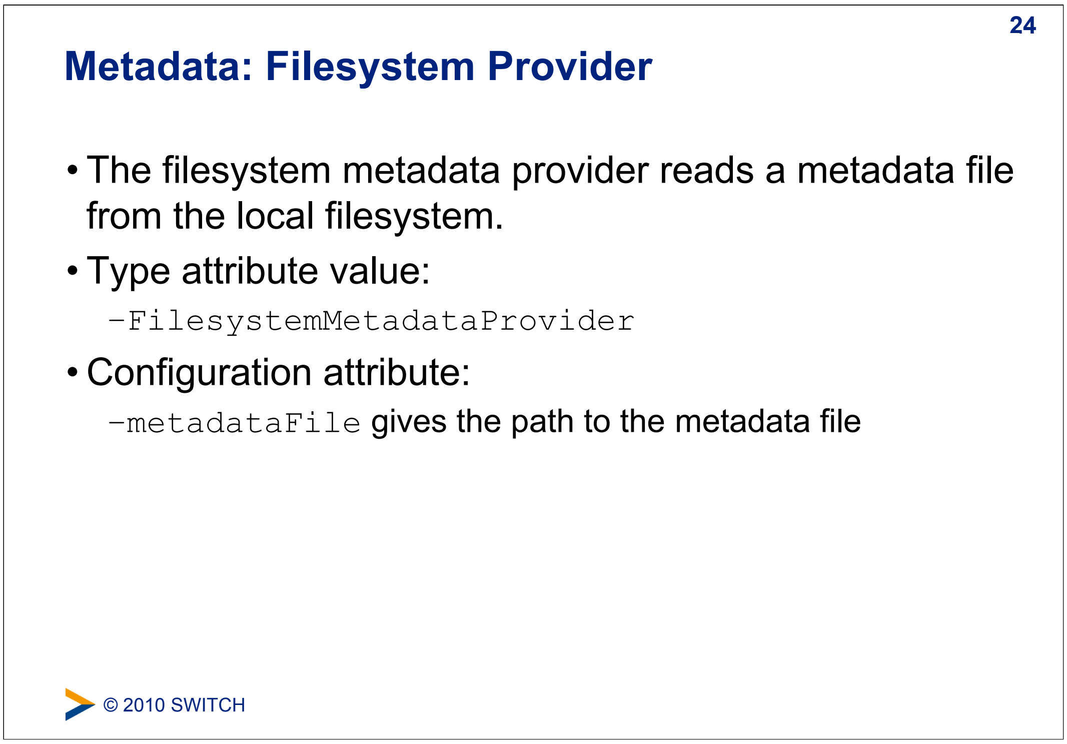
- Metadata providers are configured using `<MetadataProvider>` element
- Every metadata provider has a:
  - unique ID given by the `id` attribute
  - type given by the `xsi:type` attribute
- Each type of metadata provider has its own set of configuration options

Notes: ____________________

____

____

____



Metadata: Filesystem Provider

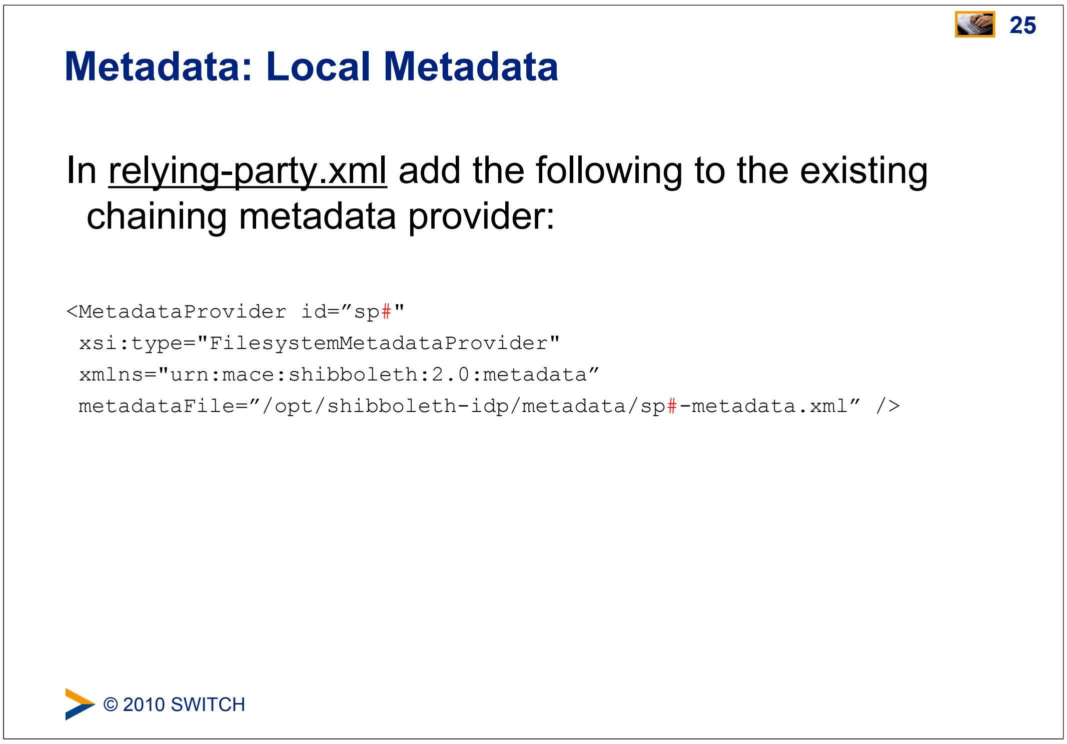
- The filesystem metadata provider reads a metadata file from the local filesystem.
- Type attribute value:
  - `FilesystemMetadataProvider`
- Configuration attribute:
  - `metadataFile` gives the path to the metadata file

Notes: ____________________

____

____

____



Metadata: Local Metadata

In relying-party.xml add the following to the existing chaining metadata provider:

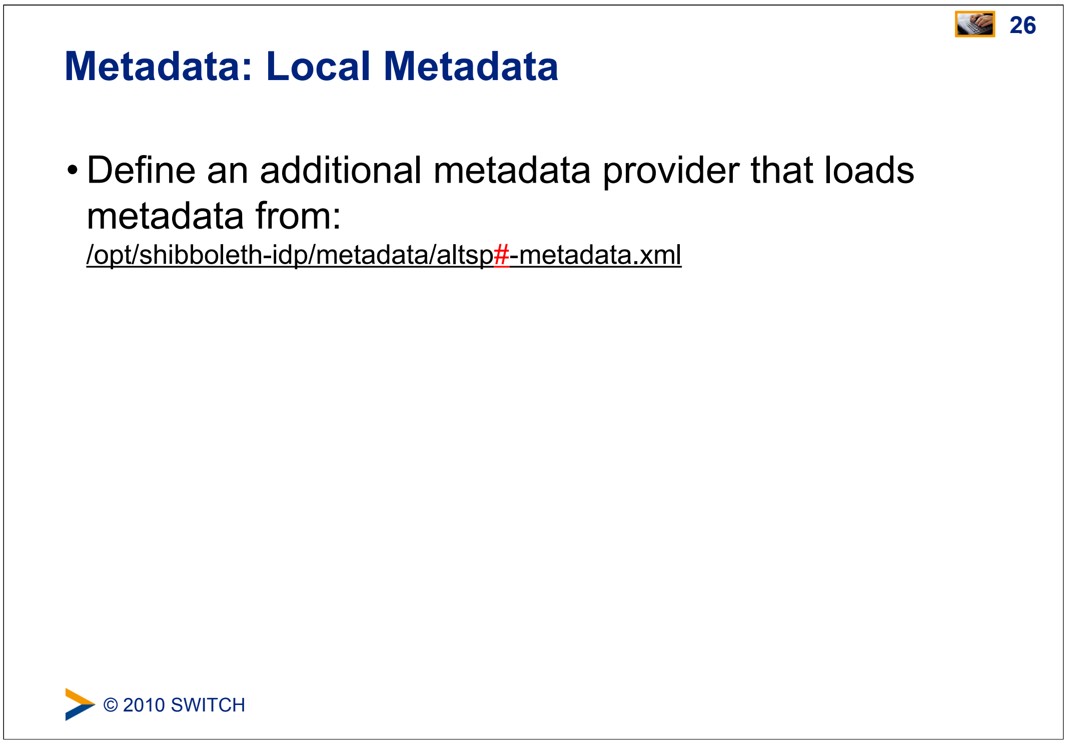
```
<MetadataProvider id=”sp#"
 xsi:type="FilesystemMetadataProvider"
 xmlns="urn:mace:shibboleth:2.0:metadata”
 metadataFile=”/opt/shibboleth-idp/metadata/sp#-metadata.xml” />
```

Notes: ____

____

____

____



Metadata: Local Metadata

- Define an additional metadata provider that loads metadata from: /opt/shibboleth-idp/metadata/altsp#-metadata.xml

Notes: ____

____

____

____

Metadata: File-backed HTTP Provider

- Loads metadata via HTTP and backs it up to a local file
- Type attribute value:
  - `FileBackedHTTPMetadataProvider`
- Configuration Attributes:
  - `metadataURL`: HTTP URL of metadata file
  - `backingFile`: location of the backup file
- In production metadata signatures should be required and validated.

Notes: ______________________________

____

____

____

Metadata: Remote Metadata

In the relying-party.xml add the following to the current chaining provider:

```
<MetadataProvider id=“testsp1”
 xsi:type="FileBackedHTTPMetadataProvider”
 xmlns="urn:mace:shibboleth:2.0:metadata”
 metadataURL=“http://testsp1.example.org/metadata.xml”
 backingFile=”/opt/shibboleth-idp/metadata/testsp1.xml”/>
```

Notes: ______________________________

____

____

____

Metadata: Remote Metadata

Define an additional metadata provider that loads metadata from http://testsp2.example.org/metadata.xml and stores it to /opt/shibboleth-idp/metadata/testsp2.xml

Notes: ____________________

____

____

____

Metadata: Watchout

The chaining metadata provider looks up relying party information in its children in the order they are defined. If two child providers load different metadata for the same entity only the first description will ever be used by the IdP. No attempt to merge the data is made.

Notes: ____________________

____

____

____