This unit describes how to detect and troubleshoot security related problems.

Unit objectives

After completing this unit, you should be able to:

- Describe common problems with WebSphere security
- Recognize symptoms of common security-related problems
- Analyze relevant log files for security messages
- Enable server tracing on relevant security components
- Analyze and interpret trace information
- Locate the security configuration files
- Use tools to validate the security configuration files

© 2007 IBM Corporation

After completing this unit, you should be able to describe common problems with WebSphere security, recognize symptoms of common security-related problems, analyze relevant log files for security messages, enable server tracing on relevant security components, analyze and interpret trace information, locate the security configuration files, and use tools to validate the security configuration files.

Review of security components and security flows

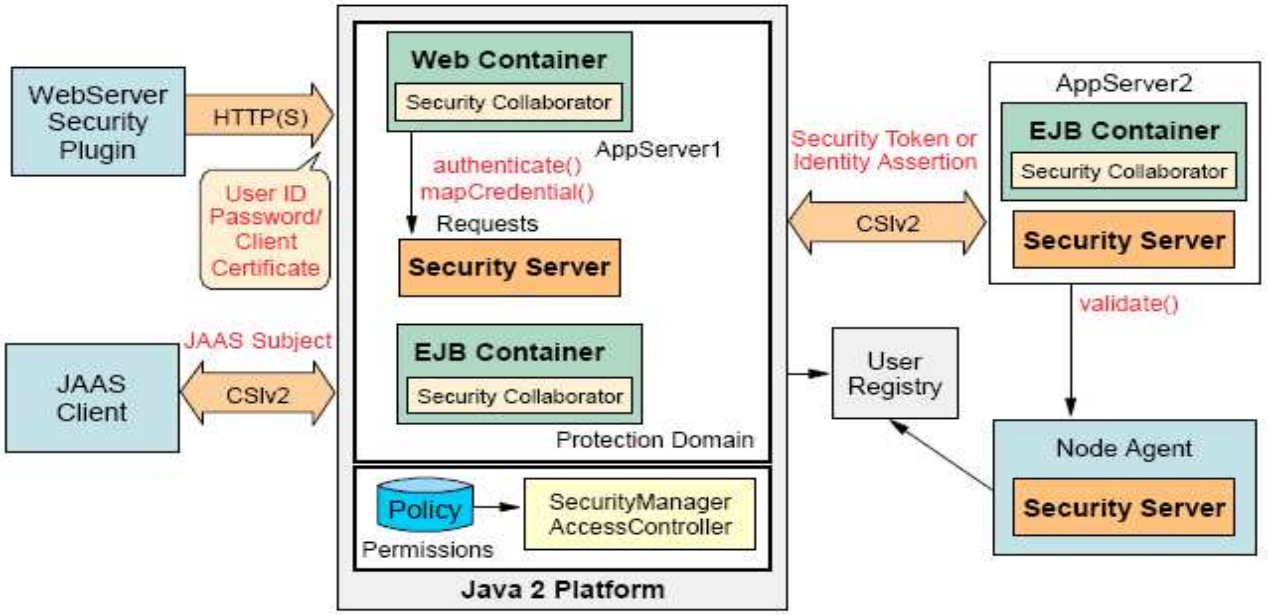
After completing this topic, you should be able to:

- List the WebSphere security components
- Describe authentication and authorization flows

© 2007 IBM Corporation

After completing this topic, you should be able to list the WebSphere security components, and describe authentication and authorization flows.

Security components--Overview



© 2007 IBM Corporation

Security components include the security server which runs in each application server, security collaborators which enforce security constraints, the web security collaborator which resides in the web container, and the EJB security collaborator which resides in the EJB container.

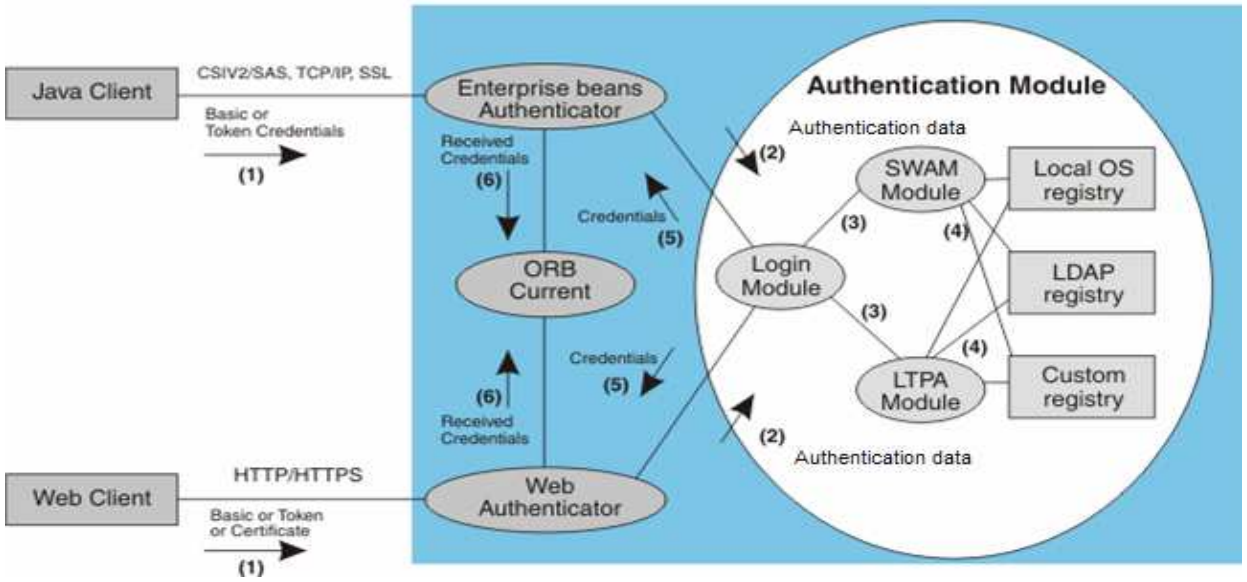
Security flows: Web browser communication

- When a Web browser sends a request to a WebSphere application, the following security interactions occur:
  1. Web user requests a Web resource that is protected by the application server
  2. The Web server receives the request and recognizes that the resource is on the application server
  3. Web server plug-in redirects the request to the Web security collaborator which performs user authentication
  4. If authentication is successful, the Web request reaches the Web container
  5. The Web security collaborator passes the following to the security server for authorization:
     - User's credentials
     - Security information contained in the deployment descriptor from EAR file
  6. For subsequent requests, authorization checks are performed either by the Web collaborator or the EJB collaborator, depending on the user's request
     - User credentials are extracted from the established security context

© 2007 IBM Corporation

When a web browsers sends a request to a WebSphere application, the resource protection is checked, the web server recognizes the resource on the server, and redirects the request to the Web security collaborator. If authentication is successful, the HTTP request reaches the web container, at which point the security server authorizes the user for the call.

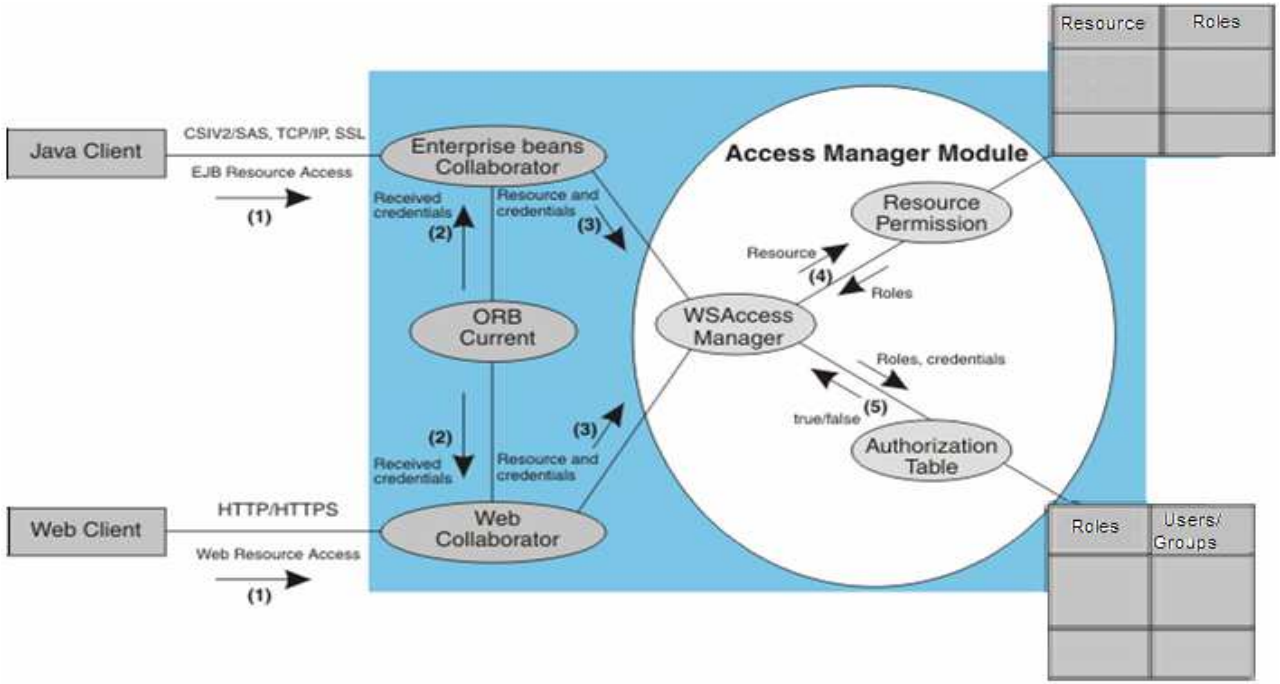
Authentication flows



© 2007 IBM Corporation

This diagram describes the process of authentication and authorization for a web client attempting to access a secured resource.

Authorization flows



© 2007 IBM Corporation

This diagram illustrates the authorization process for a web client and a Java™ client.

Security flows: Java client communication

- When a Java client interacts with a WebSphere application, the following occurs:
  1. A Java client generates a request that reaches the server side ORB
  2. The CSIv2 or IBM SAS interceptor performs authentication on the server side on behalf of the ORB, and sets the security context
  3. The server side ORB passes the request to the EJB container
  4. After submitting a request to the access-protected EJB method, the EJB container passes the request to the EJB collaborator
  5. The EJB collaborator reads the deployment descriptor from the EAR file and reads the user credentials from the security context
  6. Credentials and security information are passed to the security server, which validates user access rights and passes this information back to the collaborator
  7. After receiving a response from the security server, the EJB collaborator authorizes or denies access to the user to the requested resource

© 2007 IBM Corporation

When a java client interacts with a WebSphere appliation, the client generates a request that reaches the server side ORB and the security interceptor performs authentication on the server side on behalf of the ORB, consequently setting the security context. The server side ORB passes the request to the EJB container, which passes it to the EJB collaborator. The EJB collaborate reads the user credentials and passes it to the security server, which validates the user access rights.

Common problems and troubleshooting methods

After completing this topic, you should be able to:

- Describe what can go wrong
- Describe how to approach a security problem
- Recognize normal security messages
- Determine if a problem is related to authentication or authorization
- Determine if a problem is Web request or EJB-related
- Determine if a problem is SSL-related
- Examine a stack trace in the system log
- Enable tracing of security components
- Analyze trace information

© 2007 IBM Corporation

After completing this topic, you should be able to describe what can go wrong, describe how to approach a security problem, recognize normal security messages, determine if a problem is related to authentication or authorization, determine if a problem is Web request or EJB-related, determine if a problem is SSL-related, examine a stack trace in the system log, enable tracing of security components, analyze trace information.

What can go wrong?

- Errors trying to enable global security
  - Invalid user IDs
  - Problems accessing the User Registry
- Errors after security is enabled
  - Authentication failures
  - Authorization errors accessing a Web page
- Access and login problems after security is enabled
  - Problems trying to log in to the Administrative Console
- Errors with the SSL configuration
  - Problems accessing resources with HTTPS URLs
- Single sign-on configuration problems
  - Authentication failures
- User authorization issues
  - Problems with user/group role mappings
- Server fails to start
  - Remote user registry inaccessible
  - Node synchronization problems

© 2007 IBM Corporation

Common security errors include errors in global security, errors after security is enabled, access and login problems, errors with the SSL configuration, single sign-on configuration, user authorization, and server start failures.

Approach to troubleshooting security-related issues

- **Does the problem occur when security is disabled?**
  - A good litmus test to determine that a problem is security-related
  - Just because a problem only occurs when security is enabled does not always make it a security problem
  - More troubleshooting is necessary to ensure the problem is really security-related
- **Did security seem to initialize properly?**
  - A lot of security code executes during server initialization
  - Examine the *SystemOut.log* and *SystemErr.log* files to check for warnings and exceptions that are security-related

© 2007 IBM Corporation

Troubleshooting security related issues involves analyzing whether or not the problem occurs when security is disabled. Just because a problem only occurs when security is enabled does not always make it a security problem. You should also ensure that security initializes properly.

Normal security initialization messages

- Messages in the **SystemOut.log** indicating normal initialization:

```
SASRas        A   JSAS0001I: Security configuration initialized.
SASRas        A   JSAS0002I: Authentication protocol: CSIV2/IBM
SASRas        A   JSAS0003I: Authentication mechanism: LTPA
SASRas        A   JSAS0004I: Principal name:
DM01:389/uid=wsadmin,cn=users,dc=ibm,dc=com
SASRas        A   JSAS0005I: SecurityCurrent registered.
SASRas        A   JSAS0006I: Security connection interceptor
initialized.
SASRas        A   JSAS0007I: Client request interceptor registered.
SASRas        A   JSAS0008I: Server request interceptor registered.
SASRas        A   JSAS0009I: IOR interceptor registered.
UserRegistryI A   SECJ0136I: Custom
Registry:com.ibm.ws.security.registry.ldap.LdapRegistryImpl has been
initialized
distSecurityC I   SECJ0243I: Security service started successfully
distSecurityC I   SECJ0210I: Security enabled true
```


© 2007 IBM Corporation

Administrators should know what log messages reflect a normal startup on a server. The SystemOut messages in this example indicate normal code intialization.

Authentication or authorization problem?

- Most security problems fall under one of these two categories
- Authentication is the process of determining who the caller is
  - When authentication fails, typically this failure is related to either the
    - Authentication protocol (CSIv2 and SAS)
    - Authentication mechanism (LTPA)
    - User registry (Local OS, LDAP, Custom)
- Authorization is the process of validating that the caller has the proper authority to invoke the requested method
  - When authorization fails, this is usually related to
    - Application bindings from assembly and deployment
    - *Identity* of the caller who is accessing the method
    - *Roles* that are required by the method

© 2007 IBM Corporation

Most security problems are related to authentication or authorization. Authentication is the process of determining who the caller is, and authorization is the process of validating that the caller has the proper authority to invoke a certain method.

Is this a Web or an EJB request? (1 of 2)

- Web requests have a completely different code path than EJB requests
- Different security features exist for Web requests than for EJB requests, requiring a different body of knowledge to resolve
- For example, when using the LTPA authentication mechanism,
  - SSO is available for Web requests
  - SSO is *not* available for EJB requests
- Web requests involve HTTP header information that is *not* required by EJB requests due to the protocol differences
- Web requests involve the Web container
- EJB requests involve the EJB container

© 2007 IBM Corporation

Web requests and EJB requests have distinct security paths. Web request involve HTTP header information which EJB requests do not. Web requests involve the web container whereas EJB requests invole the EJB container. It is important to understand both request flows.

Is this a Web or an EJB request? (2 of 2)

- Secure EJB requests flow over the RMI/IIOP protocol and rely heavily on the
  - ORB component
  - Naming component
- When Workload Manager is enabled, other behavior changes in the code can be observed
- All of these components must interact closely for security to work properly

© 2007 IBM Corporation

Secure EJB requests flow over the RMI/IIOP protocol and rely on the ORB and Naming components. When Workload Manager is enabled, other differences may be seen in request behavior. Understanding the functions of the ORB and the Naming Service components can be helpful in problem determination.

Problems related to Secure Sockets Layer?

- SSL is a distinct layer of security
  - Problems are usually separate from authentication and authorization
- SSL problems are usually first-time setup problems because the configuration can be difficult
  - Each client must contain the signer certificate of the server
  - During mutual authentication, each server must contain the client's certificate
- There can be protocol differences:
  - SSLv3 versus Transport Layer Security (TLS)
- Listener port problems related to stale Interoperable Object References (IORs)
  - For example, IORs from a server that reflect the port used prior to the server's restarting

© 2007 IBM Corporation

SSL is a distinct layer of security. Problems in SSL are usually separate from authentication and authorization. Most SSL problems are related to first-time setup and configuration, including the use of IBM‘s Key Management tool. Issues included protocol differences and listener port incompatabilities.

Errors configuring SSL encrypted access

- The Java Cryptographic Extension (JCE) files were not found
  - Error when launching IKeyman

- Unable to verify MAC (message authentication code)
  - Error when the wrong keystore password is used

```
CWPKI0033E: The keystore located at
"C:/WebSphere/AppServer/profiles/profile1/etc/trust.p12" failed
to load due to the following error: Unable to verify MAC.
```

- SSL handshake failure

```
CWPKI0022E: SSL HANDSHAKE FAILURE: A signer with SubjectDN
"CN=was6host.ibm.com, O=IBM, C=US" was sent from target host:port
“10.65.49.131:9428".
```

- The certificate alias cannot be found in the keystore

© 2007 IBM Corporation

Example errors include errors launching KeyMan, such as the fact that the Java Cryptographic Exectension files are not found. An inability to verify the message authentication code signifies that the wrong keystore password is being used. An SSL handshake failure indicates that there is no trusted certificate found.

Is there a stack trace in the system log file?

- A single stack trace tells a lot about the problem
  - What code initiated the code that failed
  - What component is failing
  - Which class the failure actually came from
- Sometimes the stack trace is all that is needed to solve the problem
  - It may pinpoint the root cause
- Other times, it only gives a clue, and may be misleading
- When product support analyzes a stack trace, they may request an additional trace if it is not clear what the problem is
- You can trace several security components with different levels of detail

© 2007 IBM Corporation

As in most troubleshooting situations, the system log files should be viewed first for exceptions, errors, and, warning messages. A single stack trace tells a lot about the problem. You may also trace several security components with a certain degree of detail to further determine the cause of a security problem.

Example: SystemOut.log stack trace

- Symptom: The Deployment manager fails to start. An examination of the Dmgr's SystemOut.log file shows many exceptions and stack traces. The beginning of the first stack trace looks like the following:

```
[4/21/06 11:45:49:509 EDT] 0000000a LdapRegistryI E
SECJ0352E: Could not get the users matching the pattern
uid=wasadmin,cn=users,dc=ibm,dc=com because of the
following exception
javax.naming.CommunicationException: DM01:389.  Root
exception is java.net.UnknownHostException: DM01 at
java.net.PlainSocketImpl.connect(PlainSocketImpl.java:1
78) at java.net.Socket.connect(Socket.java:478) at
java.net.Socket.connect(Socket.java:428) at
java.net.Socket.<init>(Socket.java:335) . . .
```


© 2007 IBM Corporation

The following example stack trace shows a situations where the deployment manager cannot reach the LDAP remote host. The first step in diagnosing this issue would be to attempt to ping DM01 from the deployment manager machine.

Tracing security components

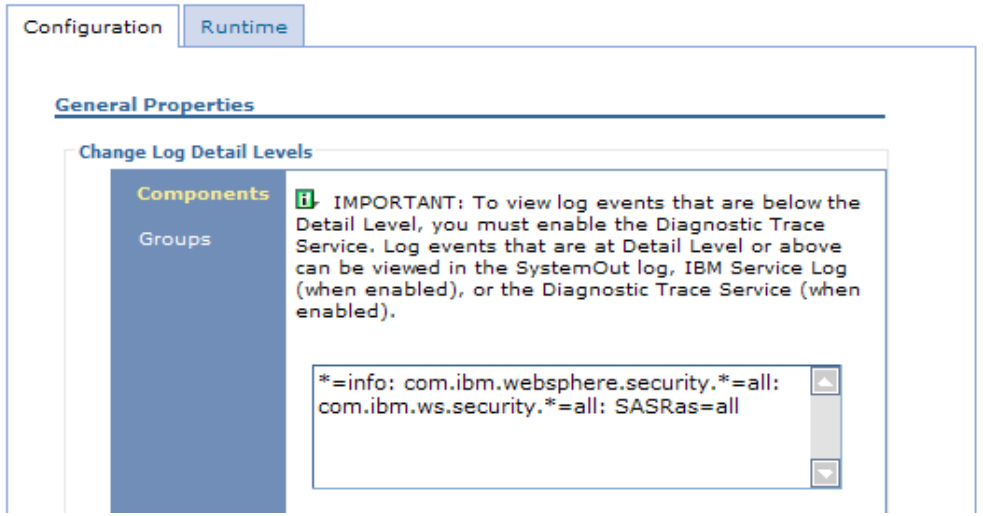
- The classes that implement WebSphere Application Server security are:
  - **com.ibm.ws.security.***
    - com.ibm.ws.security.audit.*
    - com.ibm.ws.security.auth.*
    - com.ibm.ws.security.core.*
    - com.ibm.ws.security.ejb.*
    - com.ibm.ws.security.policy.*
    - com.ibm.ws.security.registry.*
    - com.ibm.ws.security.role.*
    - com.ibm.ws.security.util.*
    - com.ibm.ws.security.web.*
  - **com.ibm.websphere.security.***
    - com.ibm.websphere.security.WSSecurityHelper
    - com.ibm.websphere.security.WebSphereSecurityPermission
  - **SASRas**

© 2007 IBM Corporation

There are various security components that can be traced such as auth, ejb, registry, and so on. The use of wildcards in trace specifications makes it easy to enable tracing on multiple components.

Enable tracing on security components

- In the Administrative Console select **Troubleshooting→Logs and Trace**
- Select the server you want to trace, **server1** or **dmgr**
- Select **Change Log Details Levels**
- Click **com.ibm.websphere.security.*** and select **all**
- Click **com.ibm.ws.security.*** and select **all**
- Click **S**



© 2007 IBM Corporation

The following example shows how to enable all the major security components for all events. Security trace tends to be significantly larger than most other kinds of traces, so performance and file size should be a consideration when deciding which trace levels to use.

Result from security components trace

- Portion of trace.log file

```
LdapRegistryI 3   Authenticating uid=wpsbind,cn=users,dc=ibm,dc=com
LdapRegistryI 3   Searching for users
LdapRegistryI >  getUsers Entry uid=wpsbind,cn=users,dc=ibm,dc=com 2
LdapRegistryI 3   pattern is full DN
LdapRegistryI >  search Entry
LdapRegistryI 3   DN: uid=wpsbind,cn=users,dc=ibm,dc=com
LdapRegistryI 3   Search scope: 0
LdapRegistryI 3   Filter: (objectclass=*)
LdapRegistryI >  getDirContext Entry
LdapRegistryI 3   enterJNDI:P=711192:O=0:CT
LdapRegistryI 3   exitJNDI:P=711192:O=0:CT
LdapRegistryI 3   Time elapsed: 2313
LdapRegistryI 3   DM01:389
LdapRegistryI 3   DM01
java.net.UnknownHostException: DM01
at java.net.PlainSocketImpl.connect(PlainSocketImpl.java:178)
```

The log shows the events leading up to the Unknown Host Exception. Apparently an attempt to access the host DM01 on port 389 (LDAP port) is failing.

© 2007 IBM Corporation

By default, when tracing is enabled for a server, the trace information is written to the trace.log file in the server‘s logs directory. In this example, without setting the trace level for the Unknown Host Exception, we would not be able to see the events that led up to the exception.

Administrative console runtime messages, log file messages, and codes

After completing this topic, you should be able to:

- Identify the different types of WebSphere and security-related messages and codes

© 2007 IBM Corporation

After completing this topic, you should be able to identify the different types of WebSphere and security-related messages and codes.

Security Association Service messages

- JSAS messages are from Security Association Service
  - Examples

JSAS0201E: [{0}] Invocation credential realm does not match target's realm: {0}. If using the SWAM authentication mechanism, you should switch to using LTPA instead for remote IIOP invocations.
Explanation: Attempting a remote invocation over IIOP using the SWAM authentication mechanism is not supported.
User Response: Retry with the LTPA authentication mechanism configured in Global Security

JSAS0202E: [{0}] Credential token expired. {1}
Explanation: The credential token associated with the user credential has expired. This typically occurs with LTPA.
User Response: Close the client and login again.

© 2007 IBM Corporation

JSAS messages are from the Security Association Service.

WebSphere Security messages

- SECJ messages are from WebSphere Security
  - Examples

SECJ0007E: Error during security initialization. The exception is {0}.
Explanation: An unexpected error occurred during security initialization.
User Response: This is a general error. Look for previous messages that may be related to the failure or a configuration problem. Enabling security debug trace for components com.ibm.ws.security.* and com.ibm.ejs.security.* may yield additional information.

SECJ0056E: Authentication failed for reason {0}
Explanation: Authentication failed with the specified reason.
User Response: Verify that the user id and password are entered correctly. Consult with the administrator of the user registry if the problem persist.

© 2007 IBM Corporation

SECJ messages are from the overall WebSphere Security component.

Web UI Security Center messages

- SECG messages are from Web UI Security Center
  - Examples

SECG0005E: An exception occurred when exporting Lightweight Third Party Authentication (LTPA) keys: The exception is {0}.
Explanation: Unable to get the Lightweight Third Party Authentication (LTPA) keys from the server.
User Response: Regenerate the keys and try the operation again.

SECG0027E: The Ignore case option is required for the Lightweight Directory Access Protocol (LDAP) directory type {0}.
Explanation: Select the Ignore case option for the LDAP directory type selected.
User Response: Enable the Ignore case option in the administrative console. Expand Security > User Registries. Click LDAP and select the Ignore case option.

© 2007 IBM Corporation

SECG messages are from the Web UI Security Center component.

Web Services Security (WS-Security) messages

- WSEC messages are from Web Services Security
  - Examples

WSEC0001E: Error trying to find Security Server. The exception is {0}.
Explanation: This exception is unexpected. The cause is not immediately known.
User Response: If the problem persists, see problem determination information on the WebSphere Application Server Support…

WSEC0007W: Server level Web Services Security configuration file {0} is not found.
Explanation: The server level Web Services Security configuration document might be corrupted or missing. The file provides the default binding configuration for Web Services Security.
User Response: If you would like to use the default bindings information, please copy ws-security.xml from the ${USER_INSTALL_ROOT}/config/templates directory.

© 2007 IBM Corporation

WSEC messages are from the Web Services Security component. Lastly, JSSL are ORB SSL Extensions messages, and WSSK are Web Services Security Kerberos messages.

CSIv2 CORBA minor codes

- The following table shows some CORBA minor codes which a client can expect to receive after running a security-related request such as authentication

Minor code name	Minor code value (Hex)	Exception type	Minor code description
AuthenticationFailed	49424300	NO_PERMISSION	See Notes
InvalidUserid	49424301	NO_PERMISSION	See Notes
InvalidPassword	49424302	NO_PERMISSION	See Notes
InvalidSecurityCredentials	49424303	NO_PERMISSION	See Notes
ServerConnectionFailed	494210A0	COMM_FAILURE	See Notes
ValueIsNull	494210B2	INTERNAL	See Notes

© 2007 IBM Corporation

CORBA exceptions are generic and indicate a problem in communication between two components. CORBA minor codes are more specific and indicate the underlying reason that a component could not complete a request. This table shows some common CORBA minor codes after running a security-related request.

Example of a CORBA exception

- Minor code name: Authentication Failed
- Minor code: 49424300

```
org.omg.CORBA.NO_PERMISSION: Caught WSSecurityContextException in
WSSecurityContext.acceptSecContext(), reason: Major Code[0] Minor
Code[0] Message[Exception caught invoking authenticateBasicAuthData
from SecurityServer for user wasadmin.

Reason: com.ibm.WebSphereSecurity.AuthenticationFailedException]

minor code: 49424300 completed: No

at com.ibm.ISecurityLocalObjectBaseL13Impl.PrincipalAuthFailReason.
map_auth_fail_to_minor_code(PrincipalAuthFailReason.java:83)

at com.ibm.ISecurityLocalObjectBaseL13Impl.CSIServerRI.receive_request
(CSIServerRI.java:1569)

at com.ibm.rmi.pi.InterceptorManager.iterateReceiveRequest
(InterceptorManager.java:739) …
```


© 2007 IBM Corporation

An example CORBA exception shows a minor code which describes an authentication failure event.

Additional tools and techniques

After completing this topic, you should be able to:

- Locate and validate security configuration files
- Track LTPA tokens
- Disable global security

© 2007 IBM Corporation

After completing this topic, you should be able to locate and validate security configuration files, track LTPA tokens, disable global security.

Security configuration files

- security.xml
  - Each profile has a copy of *security.xml* located at <WAS_HOME>\profiles\<Profile_Name>\config\cells\<Cell_Name>
  - Contains all of the security configuration information and status
    - User registry
    - Authentication mechanism
    - Many more
- ws-security.xml
  - Each application server has a copy of the *ws-security.xml* file, which defines the default binding information for Web services security
- sas.client.props
  - Each profile has a copy of *sas.clients.props* in its properties directory
  - Contains client-side properties used by Secure Association Services
- was.policy
  - Each profile has a copy of *was.policy* in its properties directory
  - Contains policy information for J2 Security

© 2007 IBM Corporation

The major configuration files related to security are security.xml, ws-security.xml, sas.client.props, and was.policy. These files should almost never be edited manually.

Tools to validate the security configuration

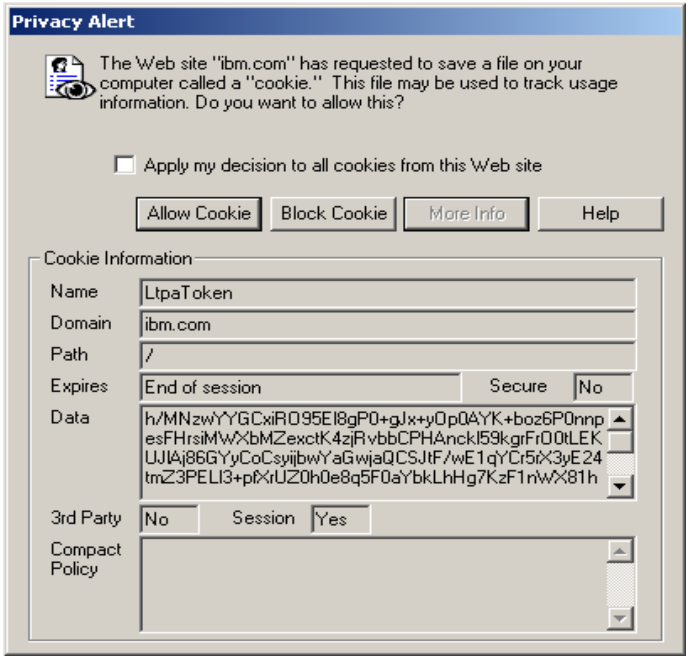
- Various tools can be downloaded from the IBM support site for checking and validating security configuration
- These tools include:
  - ACert
    - A command-line tool that checks expiration dates of all SSL certificates defined in WebSphere Application Server SSL repositories
    - The expiration dates of each certificate are displayed
  - WebSphere Security Scanning Tool (wsst)
    - A command-line tool that scans static WebSphere Application Server security configuration files to look for potential vulnerabilities
    - The tool produces an HTML report that contains
      - security configuration checks performed
      - status of each check
      - corrective action if necessary
      - a link to the information center task related to the corrective action
- LDAP Browsers/Editors
  - Very useful for working with LDAP user registries

© 2007 IBM Corporation

There are various tools from IBM support which check and validate security configuration. ACert is a command-line tool that checks expiration dates of all SSL certificates. The WebSphere Security Scanning Tool scans static security configuration files to look for potential vulnerabilities, and the LDAP Browser is useful for working directly with LDAP user registries.

Tracking LTPA tokens

- To track LTPA tokens, configure your Web browser to warn about cookies
- If you do not get a warning after a successful authentication you may have problems with
  - Domain suffix for the LTPA SSO configuration
  - Proxy server configuration



© 2007 IBM Corporation

It is useful to determine how WebSphere Application Server is managing the LTPA cookie or token. You should enable your Web browser to warn about cookies. Once you do this, your browser will inform you when WebSphere Application Server sends back an LTPA token.

Disabling global security

- Sometimes it is necessary to disable global security in order to troubleshoot security-related problems
- If the application server or deployment manager is running, use the administrative Console
  1. Select **Security → Global security**
  2. Un-select **Enable global security**
  3. Save changes to the master repository
- If the application server cannot be started, for example
  - Password of the server user ID in the user registry is expired
  - Or the user registry can not be reach for authentication
    - Disable Global Security using the command line enter
      1. <WAS_HOME>\bin\wsadmin.bat -conntype NONE
      2. wsadmin>securityoff
      3. wsadmin>quit
      4. Start server1 or dmgr

© 2007 IBM Corporation

Sometimes it is necessary to disable global security in order to troubleshoot security-related problems. This can be done through the administrative console or directly through wsadmin.

Checkpoint

1. Which two application server components have a security collaborator process?
2. What categories do most security problems fall under?
3. In which log files would you most likely find stack traces resulting from security-related exceptions?
4. Describe how you can get more detailed information about WebSphere security components written to a log file?
5. Which configuration file contains the global security information including: security status, user registry, and authentication mechanisms?

© 2007 IBM Corporation

As a checkpoint, which two application server components have a security collaborator process? What categories do most security problems fall under? In which log files would you most likely find stack traces resulting from security-related exceptions? Describe how you can get more detailed information about WebSphere security components written to a log file? Which configuration file contains the global security information including: security status, user registry, and authentication mechanisms?

Checkpoint solutions

1. The **Web container** and the **EJB container** each have a security collaborator.
2. **Authentication** and **authorization** mostly. But other categories include: SSL, Web Services, security proxies.
3. The **SystemOut.log** and **SystemErr.log** would contain stack traces resulting from security-related exceptions.
4. For each application server, node agent, and deployment manager you can enable tracing of security components at different levels of detail. This information can be written to a trace.log file.
5. The **security.xml** file contains all of the global security configuration information.

© 2007 IBM Corporation

The Web container and the EJB container each have a security collaborator. Authentication and authorization, SSL, Web Services, and security proxies. The SystemOut.log and SystemErr.log would contain stack traces resulting from security-related exceptions. For each application server, node agent, and deployment manager you can enable tracing of security components at different levels of detail. This information can be written to a trace.log file. The security.xml file contains all of the global security configuration information.

Unit summary

Having completed this unit, you should be able to:

- Describe common problems with WebSphere security
- Recognize symptoms of common security-related problems
- Analyze relevant log files for security messages
- Enable server tracing on relevant security components
- Analyze and interpret trace information
- Locate the security configuration files
- Use tools to validate the security configuration files

© 2007 IBM Corporation

Having completed this unit, you should be able to, describe common problems with WebSphere security, recognize symptoms of common security-related problems, analyze relevant log files for security messages, enable server tracing on relevant security components, analyze and interpret trace information, locate the security configuration files, and use tools to validate the security configuration files.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject= Feedback about SW5706G10_Security_Config_Probs.ppt

© 2007 IBM Corporation

You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

Approach IBM WebSphere

Access, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

EJB, Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

© 2007 IBM Corporation