COMVERSE UNIVERSITY

U-API Configuration and Administration

U-API Configuration and Administration – Objectives

By the end of this lesson, you will be able to:

- Locate important configuration files
- Start and stop the Unified API servers
- Make modifications to the Unified API configuration to change runtime behavior
- Identify the different log files created by Unified API

Cheatsheet

Use the Cheatsheet to find all the commands shown in this lesson.

Important Locations

/home/sapi/server/bin

- Scripts to startup/shutdown Unified API
- Scripts to refresh cache

/home/sapi/server/domain

- Also known as $DOMAIN_HOME
- Standard output/error logs (start_managed_sapi.log)
- Many subdirectories related to WebLogic and Unified API

/home/sapi/server/domain/SAPI_Logs

- Application Logs (SingleAPI.log, SingleAPI.log.1, and so on)

/home/sapi/server/domain/config

- WebLogic config.xml file and jdbc/jms subdirectories
- CCBSConfiguration.properties, CCBSConfiguration.xml
- Logging configuration (log4j.xml)

mShell Commands

- execute_process -g application -p sapi_admin -d start -t NI
- execute_process -g application -p sapi_managed -d start -t NI
- execute_process -g application -p sapi_managed -d shutdown
- execute_process -g application -p sapi_admin -d shutdown

Manual Start/Stop – U-API Admin Server

The official way to restart the Unified API servers is using **mShell** and the Unified Platform. Under normal circumstances, the Unified Platform restarts the Unified API server if it goes down.

To start/stop the Unified API Admin server manually:

- Log in to the U-API host with MODE=MASTER for a given cluster.
- Switch to the **apiuser** UNIX user and change to the bin directory.
- Run **start_sapi_admin.sh** or **stop_sapi_admin.sh**.
- For example:

  su - apiuser

  cd /home/sapi/server/bin

  ./start_sapi_admin.sh or **./stop_sapi_admin.sh**

The Unified Platform starts/stops the U-API server as the user id **apiuser**. The U-API servers should always be run as this user.

Manual Start/Stop – U-API Managed Server

The official way to restart the Unified API servers is using **mShell** and the Unified Platform. Under normal circumstances, the Unified Platform restarts the Unified API server if it goes down.

To start/stop the Unified API Managed server manually:

- Log in to the U-API host for a given cluster
- Switch to the **apiuser** UNIX user and change to the bin directory
- Run **start_managed_sapi.sh** or **stop_managed_sapi.sh**.
- For example:

  su - apiuser

  cd /home/sapi/server/bin

  ./start_managed_sapi.sh or **./stop_managed_sapi.sh**

The Unified Platform starts/stops the U-API server as the user id **apiuser**. The U-API servers should always be run as this user.

Health Check – Exercise

1. Open the cheatsheet.
2. Locate the SAPI Managed server health check page.
3. Enter the link.

Health Check – Exercise (Answer)

Perform a health check by going to:

http://unitip:8001/H.jsp

SAPI LBA Health Check Report

(SAPI is deployed as CONVERGED)

Database Status :

Overall health : Good.

Database Type	Database Name	Server ID	Status
BILLING_CATALOG	CTLG	1	good
UNSCALEDPC	BLUS	4	good
BILLING_CUSTOMER	CUST1	5	good
BILLING_CUSTOMER	CUST2	6	good
BILLING_CUSTOMER	CUST3	7	good
RATING_MASTER	MAIN	9	good
RATING_SLAVE	MAIN2	10	good
RATING_HISTORY	HIST	9	good
RATING_HISTORY	HIST2	10	good

Unified API Configuration

Configuration is spread across multiple files:

- **config.xml** – WebLogic configuration: JDBC/JMS resources, deployed applications: can also be modified via WL console
- **CCBSConfiguration.properties** – contains various configuration parameters related to Unified API behavior
- **CCBSReloadConfiguration.xml** – Controls how often configuration and caching information is reloaded
- **workpoint-client.properties** – Contains connection details that allow the Unified API to connect to the RH&T server (only needed in converged environments)
- **log4j.xml** – contains logging configuration

Question 1

Open the cheatsheet and answer the following questions:

1. In which directory are the configuration files located?
2. What is the user?

WebLogic config.xml (1)

Listen Address/Port

- WebLogic must be configured to listen on an interface and port
- WebLogic handles multiple protocols such as HTTP, HTTPS, T3, T3S, RMI-IIOP/EJB and SOAP over the same port by performing protocol multiplexing over their own T3 protocol.
- Important: when trying to access the WebLogic console, perform a health-check or write client applications with U-API
- The address/port are defined as child elements to the <server> XML element. The address is defined with <listen-address> and the port number is defined with <listen-port>.
- ADMIN listens on 7001; MANAGED listens on 8001 or 8051

```
<server>
    <name>ADMIN</name>
    <listen-port>7001</listen-port>
    <listen-address>sapi1.comverse.com</listen-address>
  </server>
```

WebLogic config.xml (2)

JTA Transaction Timeout

- Relevant to U-API because it defines the max. seconds that any given transaction may be active before it is rolled back
- Typically set to 500 (~8 ½ minutes).
- There should be no reason to change this under normal circumstances
- This timeout is specified as the text content of the <timeout-seconds> XML element as follows:

```
<jta>
  <timeout-seconds>500</timeout-seconds>
</jta>
```

Unified API Configuration – Exercise

1. Open the cheatsheet.
2. Locate the logging configuration file: **log4j.xml**
3. What is the log level mention in the file?

WebLogic config.xml (3)

Database Configuration (1)

- Configured as “JDBC system resources” in the config.xml
- U-API connects to MAIN, HIST, CCDB at minimum.
  - Rating Only → +ORP
  - Converged → +CTLG, BLUS, CUST1, CUST2, …
- Each “JDBC system resource” has a <jdbc-system-resource> XML element that contains the name, target server, and descriptor file name from the jdbc/ subdirectory.

```
<jdbc-system-resource>
  <name>RATING_MASTER_DB</name>
  <target>SAPI_CLUSTER</target>
  <descriptor-file-name>jdbc/RATING_MASTER_DB-5801-
jdbc.xml</descriptor-file-name>
</jdbc-system-resource>
```

WebLogic config.xml (4)

Database Configuration (2)

- For each JDBC system resource, a separate file exists in the $DOMAIN_HOME/config/jdbc/ subdirectory. This file contains additional information about the JDBC resource
  - JDBC URL
  - JDBC Driver properties
  - connection pool settings
  - JNDI names for the JDBC resource and XA parameters

```
<name>BILLING_CUST_DB</name>
  <url>jdbc:oracle:thin:@10.106.106.22:1521:cust1</url>
    <driver-
name>oracle.jdbc.xa.client.OracleXADataSource</driver-name>
    <initial-capacity>1</initial-capacity>
    <max-capacity>300</max-capacity>
    <capacity-increment>1</capacity-increment>
    <shrink-frequency-seconds>900</shrink-frequency-seconds>
    <jndi-name>jdbc/ONLINEPC</jndi-name>
```

CCBSConfiguration.properties (1)

Contains various configuration parameters related to Unified API:

- Security Related Properties
- Cybersource/ICS Configuration (Payment Gateway)
- JAX-WS Configuration (Web Services)
- Reload Manager Configuration (reloads cache/config files)
- Enumerated Data Manager Settings
- Defaults, Constraints and Message Manager Settings
- Logging settings
- Ordering settings
- Rating Control Server settings (RCS on demand)
- Transaction Timeout settings
- Trivnet Settings

CCBSConfiguration.properties (2)

The following are some of the more important properties:

- **security.server.ip** – Security Server IP address
- **security.authorization.enabled** – Controls whether API requests are authorized for a given user/security token
- **reloadMgr.enabled** – determines whether the Reload Manager is enabled (see CCBS Reload Configuration for more information)
- **logger.allowLocalLog** – defines whether a response log can be defined in the UserContext to allow client side logging
- **transaction.timeout.default** – the default transaction timeout for all API requests
- **transaction.timeout.com.comverse.api.csm.ordering.business.OrderBusiness.orderCommit** – a method to specify a custom timeout for a specific business method

CCBSReloadConfiguration.xml (1)

Controls when certain parts of U-API are refreshed:

- General configuration (CCBSConfiguration)
- Logging (LogUtil)
- Rating Control Server (RcsSocketPool)
- Constraints, Defaults and Message managers
- Enumeration manager (ExtendedDataManager)
- Extended data
- System parameters
- Various tables that change infrequently
  - BILL_CYCLE
  - INITIAL_AUT_REF
  - SERVER_DEFINTION
  - And so on

CCBSReloadConfiguration.xml (2)

- **class_name** – name of class implementing Reloadable interface
- **reload_method** – name of method implementing reload feature
- **reload_frequency** – frequent (15min), infrequent (24hr) or never
- **preload[_method]** – whether to preload on startup and method
- **pc_version_[sensitive|method]** – whether the table info should be reloaded when a new PC version (propagation) occurs and method to call that implements the reload logic

```
<reloadable>
   <name>CCBSConfiguration</name>

<class_name>com.comverse.api.framework.base.CCBSConfiguration
</class_name>
   <reload_method>reload</reload_method>
   <reload_frequency>frequent</reload_frequency>
   <preload>false</preload>
   <preload_method></preload_method>
   <pc_version_sensitive>false</pc_version_sensitive>
   <pc_version_method></pc_version_method>
</reloadable>
```

CCBSReloadConfiguration.xml (3)

If you need to change the log levels (via log4j.xml) without restarting the server, then you need to set the **reload_frequency** to **frequent** to reconfigure the logging every 15 minutes.

```
<reloadable>
   <name>LogUtil</name>
   <class_name>com.comverse.api.framework.log.LogUtil
</class_name>
   <reload_method>reload</reload_method>
   <reload_frequency>frequent</reload_frequency>
   <preload>false</preload>
   <preload_method></preload_method>
   <pc_version_sensitive>false</pc_version_sensitive>
   <pc_version_method></pc_version_method>
</reloadable>
```

workpoint-client.properties

Contains the connection information need for U-API to connect to the RH&T server

- Only relevant for Converged deployments
- **java.naming.factory.initial** – specifies initial context factory; should always be weblogic.jndi.WLInitialContextFactory
- **java.naming.provider.url** – specifies the uniform resource locator (URL) of the RH&T server
- **client.connect** = Specifies the protocol to, either EJB or XML. Always EJB when UAPI is acting as a client to RH&T

```
# Determines if the designer will connect to the EJB or MTS server,
# or use XML to communicate with a web service (good for firewalls)
client.connect = EJB

# These properties should be uncommented when using the BEA WebLogic
server.
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory
java.naming.provider.url=t3://rht1.comverse.com:8051
```

log4j.xml

Configures logging properties for the U-API business logic

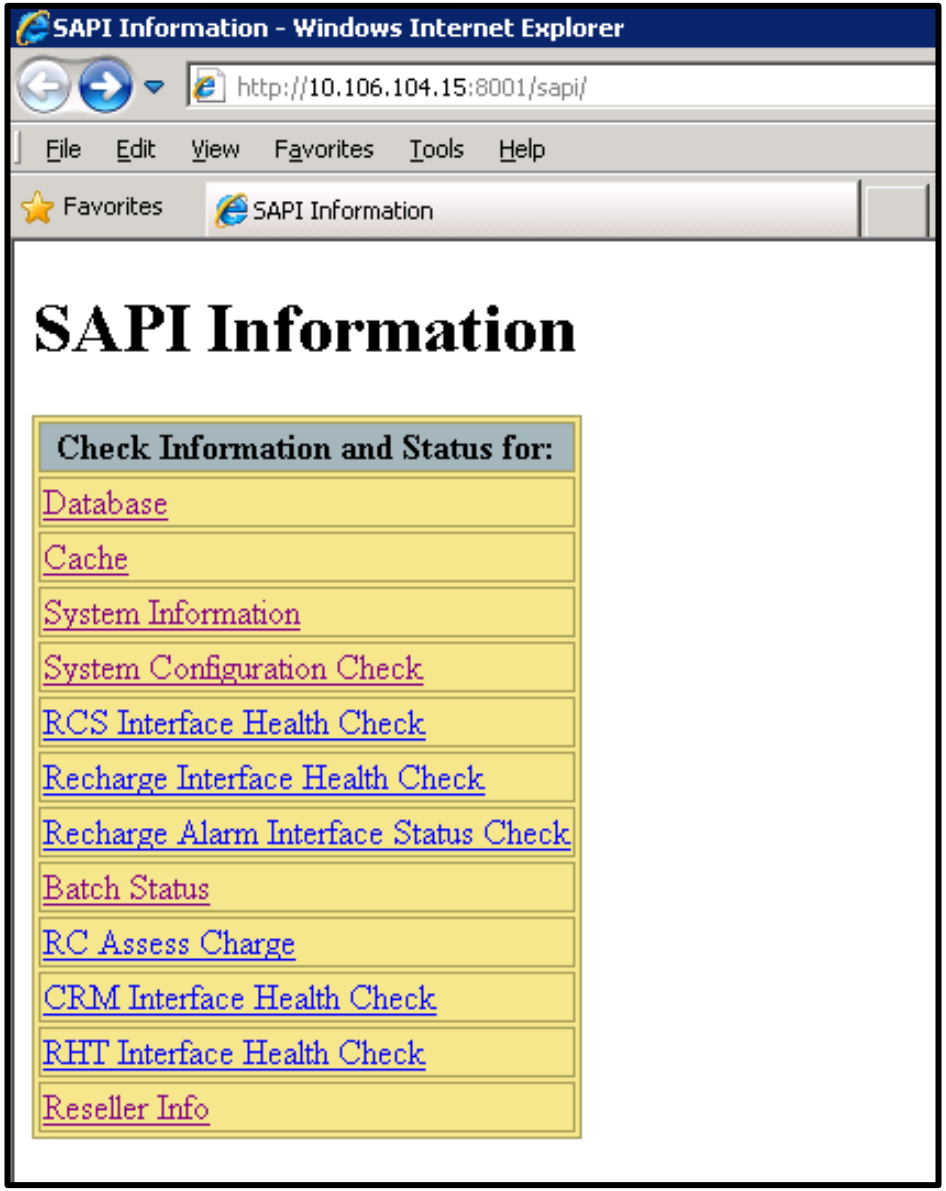
- Uses the popular log4j logging library for Java applications
- Specifies the log rotation strategy
  - Logs rotate once they reach about 10MB in size
  - SingleAPI.log contains the current log file, old log files are renamed SingleAPI.log.1, SingleAPI.log.2, and so on
- Specifies log level for different components/classes in U-API
- The example below shows how the debug level is set to **debug** for the **com.comverse.api.csm** package.

```
<logger name="com.comverse.api.csm
  <level value="debug" />
</logger>
```

Unified API Log Files

- **$DOMAIN_HOME/start_sapi_admin.log**
- **$DOMAIN_HOME/start_managed_sapi.log**
  - Created by the start_sapi_admin.sh and start_managed_sapi.sh scripts
  - Contain standard output/error (messages that would normally be printed to the console) by both WebLogic and the U-API
- **$DOMAIN_HOME/SAPI_Logs/SingleAPI.log**
  - Generated by MANAGED U-API servers
  - Contain requests/responses made by CSM, CCC, and other client applications
- **$DOMAIN_HOME/servers/{ADMIN,MS_sapi1}/logs**
  - These logs are created by WebLogic and contain administrative errors and messages

Administration



Administration – Database

SAPI Database Information

DS Name	Database Name	Database Type	Status	Max Capacity	Active Connections	Error Message
SAPI_RATING_SLAVE_DB	main	RATING_SLAVE/2	Good	200	0	
SAPI_RATING_MASTER_DB	main	RATING_MASTER	Good	200	0	
SAPI_SSA_DATA_DB	SAPI_SSA_DATA_DB	SSA_DATA	Good	15	0	
SAPI_RATING_HISTORY_DB	hist	RATING_HISTORY/2	Good	100	0	
SAPI_ASYNCH_DB	blus	ASYNC	Good	100	0	
SAPI_BILLING_UNSCALED_DB	blus	BILLING_UNSCALED	Good	100	0	
SAPI_BILLING_CATALOG_DB	ctlg	BILLING_CATALOG	Good	100	0	
SAPI_RATING_CCDB_DB	ccdb	RATING_CCDB	Good	100	0	
SAPI_ONLINE_PC_DB	cust1	ONLINEPC	Good	100	0	
SAPI_UNSCALED_DB	main	UNSCALEDPC	Good	100	0	
SAPI_RATING_VCDB_DB	main	RATING_VCDB	Good	100	0	
SAPI_BILLING_CUST_DB_3	cust1	BILLING_CUSTOMER/3	Good	100	0	

Administration – Cache

SAPI Cache Information

Reload All Cache Data

Cache Statistics

Cache Name	Hits	In Memory Hits	On Disk Hits	Misses	Object Count	Statistics Accurac
ACCESSNUMBER_CACHE	0	0	0	14	0	Best Effort
ACCOUNTCATEGORY_CACHE	0	0	0	12	0	Best Effort
ACCOUNTSEGMENT_CACHE	0	0	0	12	0	Best Effort
ACCUMULATOREXCLUSIONINCLUSION_CACHE	0	0	0	12	0	Best Effort
ACCUMULATOR_CACHE	0	0	0	17	0	Best Effort
ADDRESSSECONDARYUNITTYPE_CACHE	0	0	0	12	0	Best Effort
ADDRESSSTREETDIRECTIONAL_CACHE	0	0	0	12	0	Best Effort
ADDRESSSTREETSUFFIX_CACHE	0	0	0	12	0	Best Effort
ADDRESSTYPE_CACHE	0	0	0	14	0	Best Effort
ANGACTIONDEF_CACHE	99	99	0	251	0	Best Effort
ANNCTYPEPROMPT_CACHE	0	0	0	12	0	Best Effort

Administration – System Information

System Parameters

ParamId	ParameterName	
129	1_BONUS_MTR	0=generate one MTR per bonus awarded, 1=generate one MTR per bonus-awarding activit
10	3P_TAX_POST_EVENT	null
27	ABI_FEED	null
207	ACCESS_NOA	0 - Use Existing NOA; 1-Force NOA to National; 2 - Force NOA to International
1	ACCOUNTING_LOCATION	null
30	ACCOUNT_BILLED_DEFAULT	null
301	ACCOUNT_HOME_PAGE	null
11	ACCRUAL_REV_LEVEL	null
7003	ACCT_CAT_MUST_SYNC_NEW_ACCOUNT	null
1	ACCT_SO_DATE_ALIGN	null

Configuration Settings

Key	Value
	null
java.vendor	Sun Microsystems Inc.
java.vendor.url	http://java.sun.com/
java.vendor.url.bug	http://java.sun.com/cgi-bin/bugreport.cgi
java.vm.specification.vendor	Sun Microsystems Inc.
java.vm.specification.name	Java Virtual Machine Specification
java.vm.specification.version	1.0
java.vm.info	mixed mode
java.vm.version	11.2-b01

Administration – System Configuration Check

SAPI System Check

SAPI System Check

Information
WARNING: SYSTEM_PARAMETERS does not contain CBS.DEFAULT_RESELLER_ID
WARNING: RTNG.CALL_CIRC_SDP does not exist in SYSTEM_PARAMETERS on ONLINEPC
INFO: RATING_MASTER.2 maps to CBS_OWNER@jdbc:oracle:thin:@rating_sdp1:1521:main
WARNING: SYSTEM_PARAMETERS on RATING_MASTER.2 param RTNG.USE_GLOBAL_USSD_RESPONSE int_value (0) does not match ONLINEPC (1)
WARNING: SYSTEM_PARAMETERS on RATING_MASTER.2 param TSS.TSS_TIME_OFF int_value (1) does not match ONLINEPC (60)
WARNING: SYSTEM_PARAMETERS on RATING_MASTER.2 param BM.BALANCE_EXPIRATION_OFFSET_DAYS int_value (1) does not matcl ONLINEPC (0)
WARNING: SYSTEM_PARAMETERS on RATING_MASTER.2 param CSM.XACT_HISTORY_DEFAULT_DAYS int_value (1) does not match ONLINEPC (0)

Summary

This lesson has covered:

- Locations for important configuration files
- Procedures to start and stop the Unified API servers
- Contents of different configuration files
- Different log files related to Unified API and their content

COMVERSE
UNIVERSITY