



FortiManager v4.0 MR3
XML API Guide



April 9, 2012

02-434-166861-20120409

Copyright© 2012 Fortinet, Inc. All rights reserved. Fortinet®, FortiGate®, and FortiGuard®, are registered trademarks of Fortinet, Inc., and other Fortinet names herein may also be trademarks of Fortinet. All other product or company names may be trademarks of their respective owners. Performance metrics contained herein were attained in internal lab tests under ideal conditions, and performance may vary. Network variables, different network environments and other conditions may affect performance results. Nothing herein represents any binding commitment by Fortinet, and Fortinet disclaims all warranties, whether express or implied, except to the extent Fortinet enters a binding written contract, signed by Fortinet's General Counsel, with a purchaser that expressly warrants that the identified product will perform according to the performance metrics herein. For absolute clarity, any such warranty will be limited to performance in the same ideal conditions as in Fortinet's internal lab tests. Fortinet disclaims in full any guarantees. Fortinet reserves the right to change, modify, transfer, or otherwise revise this publication without notice, and the most current version of the publication shall be applicable.

Visit these links for more information and documentation for your Fortinet product:

Technical Documentation: <http://docs.fortinet.com>

Knowledge Base: <http://kb.fortinet.com>

Customer Service & Support: <https://support.fortinet.com>

Training Services Online Campus: <http://training.fortinet.com>



Introduction	5
Conventions	5
IP addresses	5
CLI constraints	5
Notes, Tips and Cautions	5
Typographical conventions	6
FortiManager XML APIs	7
Using the FortiManager API	8
Connecting to FortiManager Web Services	8
Enabling Web Services	8
Obtaining the WSDL file	8
Getting information from the FortiManager unit	9
addAdom	9
addDevice	10
addGroup	11
deleteAdom	11
deleteConfigRev	12
deleteDevice	13
deleteGroup	13
editAdomMembership	14
editGroupMembership	14
getAdomList	15
getDeviceList	16
getDevices	18
getConfig	19
getConfigRevisionHistory	21
getDeviceLicenseList	22
getGroupList	22
getPackageList	23
getTaskList	24
getTCLRootFile	26
listRevisionId	27
retrieveConfig	27
revertConfig	28

Working with scripts	30
createScript.....	30
deleteScript.....	31
getScript	31
getScriptLog	32
getScriptLogSummary	33
installConfig	33
runScript	34



Welcome and thank you for selecting Fortinet products for your network protection.

FortiManager v4.0 MR3 includes a Web Services interface to facilitate integration with provisioning systems. The FortiManager API is XML-based. This manual describes how to use the FortiManager API to obtain information from the FortiManager unit, run scripts to modify device configurations, and install the modified configurations on the managed devices.

This chapter contains the following topics:

- [Conventions](#)
- [IP addresses](#)
- [CLI constraints](#)
- [Notes, Tips and Cautions](#)
- [Typographical conventions](#)

Conventions

Fortinet technical documentation uses the conventions described below.

IP addresses To avoid publication of public IP addresses that belong to Fortinet or any other organization, the IP addresses used in Fortinet technical documentation are fictional and follow the documentation guidelines specific to Fortinet. The addresses used are from the private IP address ranges defined in RFC 1918: Address Allocation for Private Internets, available at ietf.org/rfc/rfc1918.txt?number-1918.

CLI constraints CLI constraints, such as `<address_ipv4>`, indicate which data types or string patterns are acceptable input for a given parameter or variable value. CLI constraint conventions are described in the CLI Reference document for each product.

Notes, Tips and Cautions Fortinet technical documentation uses the following guidance and styles for notes, tips and cautions.



Tip: Highlights useful additional information, often tailored to your workplace activity.



Note: Also presents useful information, but usually focused on an alternative, optional method, such as a shortcut, to perform a step.



Caution: Warns you about commands or procedures that could have unexpected or undesirable results including loss of data or damage to equipment.

Typographical conventions Fortinet documentation uses the following typographical conventions:

conventions

Table 1: Typographical conventions in Fortinet technical documentation

Convention	Example
<i>Button, menu, text box, field, or check box label</i>	From <i>Minimum log level</i> , select <i>Notification</i> .
CLI input	<pre>config system dns set primary <address_ipv4> end</pre>
CLI output	<pre>FGT-602803030703 # get system settings comments : (null) opmode : nat</pre>
<i>Emphasis</i>	HTTP connections are <i>not</i> secure and can be intercepted by a third party.
<i>File content</i>	<pre><HTML><HEAD><TITLE>Firewall Authentication</TITLE></HEAD> <BODY><H4>You must authenticate to use this service.</H4></pre>
Hyperlink	Visit the Fortinet Technical Support web site, support.fortinet.com .
Keyboard entry	Type a name for the remote VPN peer or client, such as <code>Central_Office_1</code> .
<i>Navigation</i>	Go to <i>VPN > IPSEC > Auto Key (IKE)</i> .
Publication	For details, see the FortiManager Administration Guide .



This chapter lists the available FortiManager XML APIs.

- addAdom
- addDevice
- addGroup
- deleteAdom
- deleteConfigRev
- deleteDevice
- deleteGroup
- editAdomMembership
- editGroupMembership
- getAdomList
- getConfig
- getConfigRevisionHistory
- getDeviceLicenseList
- getDeviceList
- getDevices
- getGroupList
- getPackageList
- getTaskList
- getTCLRootFile
- listRevisionId
- retrieveConfig
- retrieveConfig
- createScript
- deleteScript
- getScript
- getScriptLog
- getScriptLogSummary
- installConfig
- runScript



The FortiManager API enables you to configure managed FortiGate devices through a Web Services interface. You can obtain information, create and run FortiOS CLI scripts on the FortiManager database, and then install the changes on FortiGate units.

The following topics are included in this section:

- [Connecting to FortiManager Web Services](#)
- [Getting information from the FortiManager unit](#)
- [Working with scripts](#)

Connecting to FortiManager Web Services

To start working with Web Services on your FortiManager unit, enable Web Services and obtain the Web Services Description Language (WSDL) file that defines the XML requests you can make and the responses that FortiManager can provide.

Enabling Web Services

You must enable Web Services on the network interfaces to which Web Services clients will connect.

To enable Web Services on an interface - web-based manager

- 1 Go to *System Settings > Network > Interface*.
- 2 Select the *Edit* icon for the interface that you want to use.
- 3 In the *Administrative Access* section, select *Web Service*.
- 4 Select *OK*.

To enable Web Services on an interface - CLI

- 1 Enter the following CLI commands:

```
config fmsystem interface
  edit <port>
    set allowaccess webservice
  end
```

where <port> is the network interface that you want to use for Web Services.

The `allowaccess` command should also include the other types of administrative access that you want to permit. For example, to allow HTTPS, SSH, and Web Services, enter the command `set allowaccess https ssh webservice`.

The FortiManager unit handles Web Services requests on port 8080.

Obtaining the WSDL file

You can download the WSDL file directly from the URL `https://<ip_address>:8080/`.

You can also use the web-based manager to download the WSDL file. Go to *System Settings > Advanced > Advanced Settings* and select the *Download WSDL file* icon.

By using a web testing tool such as SoapUI, you can get information from the FortiManager.

Getting information from the FortiManager unit

To work with your managed devices, you need to obtain information from the FortiManager unit, such as:

- the list of ADOMs
- information of the managed devices
- information about individual devices
- the current configuration of devices, according to the database
- the revision history of devices.

addAdom Use this request to add an ADOM to the FortiManager unit.

Table 2: addAdom request fields

<userID>	Administrator ID, e.g. admin.
<password>	Enter your admin password.
<name>	The host name of the device.
<version>	Enter the firmware version.
<mr>	Enter the Major Release version.
<isBackupMode>	Use for Backup Mode ADOM.
<VPNManagement>	Use for VPN console ADOM.
<deviceSNVdom>	Enter the device serial number for the VDOM.
<SN>	Enter the device serial number.
<vdomName>	Enter the name of the VDOM.
<vdomID>	Enter the VDOM identifier.

Example request:

```
<r20:addAdom>
  <servicePass>
    <userID?></userID>
    <password?></password>
  </servicePass>
  <name?></name>
  <version?></version>
  <mr?></mr>
  <isBackupMode?></isBackupMode>
  <VPNManagement?></VPNManagement>
  <deviceSNVdom>
    <SN?></SN>
    <vdomName?></vdomName>
    <vdomID?></vdomID>
  </deviceSNVdom>
  <deviceIDVdom>
```

```
<ID>?</ID>
</r20:addAdom>
```

addDevice Use this request to add a device to the FortiManager unit.

Table 3: addDevice request fields

<password>	Administrator password.
<userID>	Administrator ID, e.g. <i>admin</i> .
<adom>	Enter the name of the ADOM, or if the ADOM status is disabled, then enter <i>root</i> .
<ip>	Enter the IP address of the device.
<autod>	Select if you want to enable auto discovery. The default value is <i>False</i> . Select the value <i>unreg</i> to promote an unregistered device.
<deviceType>	Select the type of device. The device type can be: FortiGate, FortiSwitch, FortiCarrier, FortiMail, or FortiAnalyzer.
<name>	The host name of the device.
<adminUser>	Enter your admin username.
<password>	Enter your admin password.
<description>	Enter the description of the device.
<waitTask>	Enter either <i>True</i> (default) or <i>False</i> . If the value is <i>False</i> , the result will be based on the task list, and in the response the <i>taskid</i> will be displayed.

Example request:

```
<r20:addDevice>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <adom></adom>
  <ip></ip>
  <autod></autod>
  <deviceType>?</deviceType>
  <name>FortiGate</name>
  <adminUser>admin</adminUser>
  <password></password>
  <description>FortiGate</description>
  <waitTask></waitTask>
</r20:addDevice>
```

The response is a series of <return> tags, each containing information about the device.

Table 4: addDevice response fields

<error msgs>	Error messages indicate: <ul style="list-style-type: none"> • 101: “device IP cannot be empty”. • 102: “name and admin user must be entered.”. • 103: “addDevice auto discovery mode is not supported yet.”. • 104: “add device by IP (%s) in ADOM (%d) failed”. • 0: “Add device added successfully”. • 0: “Read taskid to get the addDevice result.”.
<devid>	N/A
<taskid>	Indicates the task id number. If the <waitTask> was False, then the task id is displayed.

addGroup Use this request to add a group to the FortiManager unit.

Table 5: addgroup request fields

<userID>	Administrator ID, e.g. <i>admin</i> .
<password>	Administrator password.
<adom>	Enter the name of the ADOM, or if the ADOM status is disabled, then enter <i>root</i> .
<name>	The host name of the device.
<description>	Enter the group description.
<deviceSN>	Enter the device serial number.
<deviceID>	Enter the device identifier.
<groupName>	Enter the group name.
<groupID>	Enter the group identifier.

Example request:

```
<r20:addGroup>
  <servicePass>
    <userID?></userID>
    <password?></password>
  </servicePass>
  <adom?></adom>
  <name?></name>
  <description?></description>
  <deviceSN?></deviceSN>
  <deviceID?></deviceID>
  <groupName?></groupName>
  <groupID?></groupID>
</r20:addGroup>
```

deleteAdom Use this request to delete an ADOM from the FortiManager unit.

Table 6: deleteAdom request fields

<userID>	Administrator ID, e.g. adom.
<password>	Administrator password.
<adomName>	Enter the name of the ADOM.
<adomOid>	Enter the ADOM object identifier (OID).

Example request:

```
<r20:deleteAdom>
  <servicePass>
    <userID?</userID>
    <password?</password>
  </servicePass>
  <adomName?</adomName>
  <adomOid?</adomOid>
</r20:deleteAdom>
```

deleteConfigRev Use this request to delete a configuration revision defined on the FortiManager unit. Only an administrator with the Super_User profile can run this command.

Table 7: deleteConfigRev request fields

<password>	Administrator password.
<userID>	Administrator ID, e.g. admin.
<devId>	Enter the device ID.
<serialNumber>	Enter the serial number of the device.
<revName>	Enter the Revision name. You can get this in the Revision History section in the GUI.
<revId>	Enter the Revision id.

Example request:

```
<r20:deleteConfigRev>
  <servicePass>
    <password?</password>
    <userID?</userID>
  </servicePass>
  <devId?</devId>
  <serialNumber?</serialNumber>
  <revName?</revName>
  <revId?</revId>
</r20:deleteConfigRev>
```

The response indicates if the procedure was successful or not.

Table 8. deleteConfigRev response fields

<error msg>	Indicates if the deletion was successful or if it failed.
--------------------------	---

deleteDevice Use this request to delete a device defined on the FortiManager unit.

Table 9: deleteDevice request fields

<password>	Administrator password.
<userID>	Administrator ID, e.g. admin.
<devId>	Enter the device id number.
<serialNumber>	Enter the serial number of the device.

Example request:

```
<r20:deleteDevice>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <devId>468985</devId>
  <serialNumber>FG3K8A3407600241</serialNumber>
</r20:deleteDevice>
```

The response indicates if the device was deleted successfully or if the procedure failed.

Table 10: deleteDevice response fields

<errorCode>	Indicates the error code number.
<errorMsg>	Indicates if the device was successfully deleted or not.

Example response:

```
<ns3:deleteDeviceResponse>
  <errorMsg>
    <errorCode>0</errorCode>
    <errorMsg>Delete device ID 468985 successfully</errorMsg>
  </errorMsg>
</ns3:deleteDeviceResponse>
```

deleteGroup Use this request to delete a group from the FortiManager unit.

Table 11: deleteGroup request fields

<userID>	Administrator ID, e.g. admin.
<password>	Administrator password.
<adom>	Enter the name of the ADOM, or if the ADOM status is disabled, then enter <code>root</code> .
<name>	The host name of the device.
<grpID>	Enter the group identifier.

Example request:

```
<r20:deleteGroup>
  <servicePass>
    <userID?></userID>
    <password?></password>
  </servicePass>
  <adom?></adom>
  <name?></name>
  <grpId?></grpId>
</r20:deleteGroup>
```

editAdomMembership Use this request to edit the ADOM membership.

Table 12: editAdomMembership request fields

<userID>	Administrator ID, e.g. admin.
<password>	Enter your admin password.
<name>	The host name of the device.
<version>	Enter the firmware version.
<mr>	Enter the Major Release version.
<isBackupMode>	Use for Backup Mode ADOM.
<VPNManagement>	Use for VPN console ADOM.
<deviceSNVdom>	Enter the device serial number for the VDOM.
<SN>	Enter the device serial number.
<vdomName>	Enter the name of the VDOM.
<vdomID>	Enter the VDOM identifier.

Example request:

```
<r20:editAdomMembership>
  <servicePass>
    <userID?></userID>
    <password?></password>
  </servicePass>
  <name?></name>
  <version?></version>
  <mr?></mr>
  <isBackupMode?></isBackupMode>
  <VPNManagement?></VPNManagement>
  <addDeviceSNVdom>
    <SN?></SN>
    <vdomName?></vdomName>
    <vdomID?></vdomID>
  </addDeviceSNVdom>
</r20:editAdomMembership>
```

editGroupMembership Use this request to edit group membership.

Table 13: editGroupMembership request fields

<userID>	Administrator ID, e.g. admin.
<password>	Enter your admin password.
<adom>	Enter the name of the ADOM, or if the ADOM status is disabled, then enter <code>root</code> .
<name>	The host name of the device.
<grpID>	Enter the group identifier.
<addDeviceSNList>	Enter the device serial number list to add.
<addDeviceIDList>	Enter the device identifier list to add.
<delDeviceSNList>	Enter the device serial number list to delete.
<delDeviceIDList>	Enter the device identifier list to delete.
<addGroupNameList>	Enter the group name list to add.
<addGroupIDList>	Enter the group identifier list to add.
<delGroupNameList>	Enter the group name list to delete.
<delGroupIDList>	Enter the group identifier list to delete.

Example request:

```
<r20:editGroupMembership>
  <servicePass>
    <userID?></userID>
    <password?></password>
  </servicePass>
  <adom?></adom>
  <name?></name>
  <grpId?></grpId>
  <addDeviceSNList?></addDeviceSNList>
  <addDeviceIDList?></addDeviceIDList>
  <delDeviceSNList?></delDeviceSNList>
  <delDeviceIDList?></delDeviceIDList>
  <addGroupNameList?></addGroupNameList>
  <addGroupIDList?></addGroupIDList>
  <delGroupNameList?></delGroupNameList>
  <delGroupIDList?></delGroupIDList>
</r20:editGroupMembership>
```

getAdomList Use this request to get a list of the ADOMs defined on the FortiManager unit. Only an administrator with the Super_User profile can run this command.

Table 14: getAdomList request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin

Example request:

```
<r20:getAdomList>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
</r20:getAdomList>
```

The response is a series of <return> tags, each containing information about an ADOM.

Table 15: getAdomList response fields

<description>	Optional description.
<name>	ADOM name.
<oid>	Object identifier for the ADOM.

Example response:

```
<ns3:getAdomListResponse>
  <return>
    <oid>3</oid>
    <name>root</name>
    <description/>
  </return>
  <return>
    information about another ADOM
  </return>
</ns3:getAdomListResponse>
```

getDeviceList Use this request to get summary information about the managed devices, optionally limited to a particular ADOM.

Table 16: getDeviceList request fields

<password>	Administrator password.
<userID>	Administrator ID, e.g. admin.
<adom>	Enter an ADOM name to list only the devices in that ADOM. If <adom> is empty or does not match an ADOM name, the response lists all devices.
<mgmtMode>	Support for unregistered devices.

Example request:

```
<r20:getDeviceList>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <adom></adom>
```

```
</r20:getDeviceList>
```

The response is a series of <return> tags, each containing information about a device.

Table 17: getDeviceList response fields

<firmware>	One of: FortiGate, FortiCarrier, FortiAnalyzer, FortiMail, or FortiSwitch.
<firmwareVersion>	Major software version number, 4, for example.
<buildNum>	Software build number, 65, for example.
<description>	Device description from FortiManager database.
<hostname>	Device host name.
<IPSContract>	FortiGuard IPS definitions version and last update time, 2.00461(2008-12-08 11:23), for example.
<antiVirusContract>	AV contract and expiry date, 8.00631(2009-02-15 14:27), for example.
<platform>	Platform name for device, FortiGate-1000A, for example.
<sn>	Serial number of device.
<ip>	IP address of device network interface from which response was received.

Example response:

```
<ns3:getDeviceListResponse>
  <return>
    <firmware>FortiGate</firmware>
    <firmwareVersion>4</firmwareVersion>
    <buildNum>68</buildNum>
    <description/>
    <hostname>FG30002801030058</hostname>
    <IPSContract>2.00542(2008-09-04 23:08)</IPSContract>
    <antiVirusContract>8.00631(2008-01-15 14:27)
      </antiVirusContract>
    <platform>FortiGate-3000</platform>
    <sn>FG30002801030058</sn>
    <ip>172.20.120.134</ip>
  </return>
  <return>
    another device
  </return>
</ns3:getDeviceListResponse>
```

getDevices Use this request to get information about specific managed devices, identified by serial number or device ID. You can obtain device ID values by using the `execute dmserver showdev` CLI command.

If you want information about the device's configuration, see “getConfig” on page 19.

Table 18: getDevices request fields

<password>	Administrator password.
<userID>	Administrator ID, e.g. admin.
<serialNumbers>	Serial number of the device. This is the secondary identifier. You can enter multiple <serialNumbers> fields.
<devIds>	Device ID. This is the primary device identifier. You can omit this field and use <serialNumber> instead. You can enter multiple <DevIds> fields.
<mgmtMode>	Support for unregistered devices.

Example request:

```
<r20:getDevices>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <serialNumbers>FGT1002801021024</serialNumbers>
  <serialNumbers>FGT50B3G06500085</serialNumbers>
</r20:getDevices>
```

The response is a series of <return> tags, each containing information about a device.

Table 19: getDevices response fields

<firmware>	One of: FortiGate, FortiCarrier, FortiAnalyzer, FortiMail, or FortiSwitch.
<firmwareVersion>	Major software version number, 4, for example.
<buildNum>	Software build number, 65, for example.
<description>	Device description from FortiManager database.
<hostname>	Device host name.
<IPSCContract>	FortiGuard IPS definitions version and last update time, 2.00461(2008-12-08 11:23), for example.
<antiVirusContract>	AV contract and expiry date, 8.00631(2009-02-15 14:27), for example.
<platform>	Platform name for device, FortiGate-1000A, for example.
<sn>	Serial number of device.
<ip>	IP address of device network interface from which response was received.

Example response:

```

<ns3:getDevicesResponse>
  <return>
    <firmware>FortiGate</firmware>
    <firmwareVersion>3.00</firmwareVersion>
    <buildNum>650</buildNum>
    <description/>
    <hostname>Dev3</hostname>
    <IPSContract>2.442(2007-11-08 11:23)</IPSContract>
    <antiVirusContract>8.368(2007-11-15
      13:59)</antiVirusContract>
    <platform>FortiGate-100</platform>
    <sn>FGT1002801021024</sn>
    <ip>172.20.120.126</ip>
  </return>
  <return>
    <firmware>FortiGate</firmware>
    <firmwareVersion>4</firmwareVersion>
    <buildNum>59</buildNum>
    <description/>
    <hostname>FGT50B3G06500085</hostname>
    <IPSContract/>
    <antiVirusContract/>
    <platform>Fortigate-50B</platform>
    <sn>FGT50B3G06500085</sn>
    <ip>172.20.120.127</ip>
  </return>
</ns3:getDevicesResponse>

```

getConfig Use this request to retrieve a particular revision of the device configuration from the device database.

Table 20: getConfig request fields

<password>	Administrator password.
<userID>	Administrator ID, e.g. admin.
<serialNumber>	Serial number of the device. This device identifier is secondary to <devId>.
<devId>	The Device ID. This is the primary device identifier. You can omit this field and use <serialNumber> instead.
<checkinUser>	The userID of the administrator who saved this revision.
<revisionNumber>	The revision that you want to view. Use a negative number to retrieve the latest revision.

Example request:

```
<r20:getConfig>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <serialNumber>FGT1002801021024</serialNumber>
  <!-- <devId></devId> -->
  <revisionNumber>3</revisionNumber>
</r20:getConfig>
```

The response is a <return> field containing the device configuration and other information.

Table 21: getConfig response fields

<branchPoint>	The firmware build number, except for some special branch builds.
<checkinDate>	Date and time (UTC) when this revision was installed to the device.
<checkinUser>	The userID of the administrator who installed this revision.
<content>	The device configuration file contents.
<oid>	The devID for the device.
<osVersion>	Version of device operating system, 4, for example for FortiOS 4.0.
<platform>	Platform name for the device, FortiGate-1000A, for example.
<revisionNum>	Configuration revision ID.
<serialNumber>	The serial number of the device.

Example response

```
<ns3:getConfigResponse>
  <return>
    <branchPoint>650</branchPoint>
    <checkinDate>2009-02-02T19:37:39Z</checkinDate>
    <checkinUser>admin</checkinUser>
    <content>
      ... configuration file content ...
    </content>
  </return>
  <message/>
    <oid>109</oid>
    <osVersion>3.00</osVersion>
    <platform>FortiGate-100</platform>
    <revisionNum>3</revisionNum>
    <serialNumber>FGT1002801021024</serialNumber>
  </return>
</ns3:getConfigResponse>
```

getConfigRevisionHistory Use this request to retrieve multiple revisions of the device configuration from the device database. You can retrieve based on revision numbers or check-in times.

Table 22: getConfigRevisionHistory request fields

<password>	Administrator password.
<userID>	Administrator ID, e.g. admin.
<serialNumber>	Serial number of the device. This device identifier is secondary to <devId>.
<devId>	The Device ID. This is the primary device identifier. You can omit this field and use <serialNumber> instead.
<checkinUser>	Optionally, specify the userID of the administrator who saved this revision.
<minCheckinDate>	Optionally, specify the earliest revision check-in time to retrieve. Use with <maxCheckinDate>.
<maxCheckinDate>	Optionally, specify the latest revision check-in time to retrieve. Use with <minCheckinDate>.
<minRevisionNumber>	Optionally, specify the first revision to retrieve. Use with <maxRevisionNumber>.
<maxRevisionNumber>	Optionally, specify the last revision to retrieve. Use with <minRevisionNumber>.

Example request:

```
<r20:getConfigRevisionHistory>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <serialNumber>FGT1002801021024</serialNumber>
  <!-- <devId>?</devId> -->
  <!-- <checkinUser></checkinUser> -->
  <!-- <minCheckinDate></minCheckinDate> -->
  <!-- <maxCheckinDate></maxCheckinDate> -->
  <!-- <minRevisionNumber>1</minRevisionNumber> -->
  <!-- <maxRevisionNumber>?</maxRevisionNumber> -->
</r20:getConfigRevisionHistory>
```

The response is a series of <return> fields containing the device configuration and other information.

Table 23: getConfigRevisionHistory response fields

<branchPoint>	The firmware build number, except for some special branch builds.
<checkinDate>	Date and time (UTC) when this revision was installed to the device.
<checkinUser>	The userID of the administrator who installed this revision.
<content>	The device configuration file contents.
<oid>	The devID for the device.
<osVersion>	Version of device operating system, 4, for example for FortOS 4.0.

Table 23: getConfigRevisionHistory response fields (Continued)

<platform>	Platform name for the device, FortiGate-1000A, for example.
<revisionNum>	Configuration revision ID.
<serialNumber>	The serial number of the device.

Example response:

```
<ns3:getConfigRevisionHistoryResponse>
  <return>
    <branchPoint>650</branchPoint>
    <checkinDate>2009-02-02T19:37:39Z</checkinDate>
    <checkinUser>admin</checkinUser>
    <content>
      ... configuration file content - newest retrieved revision
      ...
    </content>
    <message/>
      <oid>109</oid>
      <osVersion>3.00</osVersion>
      <platform>FortiGate-100</platform>
      <revisionNum>3</revisionNum>
      <serialNumber>FGT1002801021024</serialNumber>
    </return>
  <return>
    ... information about preceding revision ...
  </return>
```

getDeviceLicense List Use this request to obtain a list of device licenses.

Table 24: getDeviceLicenseList

<userID>	Administrator ID, e.g. admin.
<password>	Enter your admin password.

Example request:

```
<r20:getDeviceLicenseList>
  <servicePass>
    <userID>?</userID>
    <password>?</password>
  </r20:getDeviceLicenseList>
```

getGroupList Use this request to obtain a list of device groups, optionally limited to a particular ADOM.

Table 25: getGroupList request fields

<password>	Administrator password.
-------------------------	-------------------------

Table 25: getGroupList request fields (Continued)

<userID>	Administrator ID, e.g. admin.
<adom>	Enter an ADOM name to list only the device groups in that ADOM. If <adom> is empty or does not match an ADOM name, the response lists all device groups.

Example request:

```
<r20:getGroupList>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <adom></adom>
</r20:getGroupList>
```

The response is a series of <return> fields, each listing one group, in ascending order of object identifier (OID). Both built-in groups, like All FortiGate, and user-defined groups are listed.

Table 26: getGroupList response fields

<name>	Group name.
<oid>	Object identifier for the group.

Example response:

```
<ns3:getGroupListResponse>
  <return>
    <oid>118</oid>
    <name>Group1</name>
  </return>
  <return>
    ... another group ...
  </return>
```

getPackageList Use this request to retrieve a package list.

Table 27: getPackageList request fields

<userID>	Administrator ID, e.g. admin.
<password>	Enter your admin password.
<adom>	Enter the name of the ADOM, or if the ADOM status is disabled, then enter <code>root</code> .

Example request:

```
<r20:getPackageList>
  <servicePass>
    <userID>?</userID>
    <password>?</password>
  </servicePass>
```

```
<adom>?</adom>
</r20:getPackageList>
```

getTaskList Use this request to get a list of tasks as defined on the FortiManager unit. Only an administrator with the Super_User profile can run this command.

Table 28: getTaskList request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<adom>	ADOM number
<taskId>	Enter the number for the task list.

Example request:

```
<r20:getTaskList>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <adom></adom>
  <taskId>1</taskId>
</r20:getTaskList>
```

The response is a series of <return> tags, each containing information about a task.

Table 29: getTaskList response fields

<errorMsg>	Indicates if the task was successful or if it failed.
<errorCode>	Indicates the error code.
<taskId>	Enter the number of the task.
<taskList>	Following variables provide details of the tasklist.
	<source> - indicates the source of the task: 0 - device manager 1 - security console 2 - copy global object 3 - install configuration 4 - script execution 5 - system checkpoint 6 - import device policy 7 - install ems global policy
	<description> - describes the list.
	<userID> - shows the user id.

Table 29: getTaskList response fields (Continued)

	<p><status> - indicates the status of the task: 1 - running 2 - cancelling 3 - cancelled 4 - done 5 - error 6 - aborting 7 - aborted</p>
	<p><statTime> - indicates the time the task list started.</p>
<deviceList>	
	<p><devName>: name of the device host.</p>
	<p><ip>: IP address of the device.</p>
	<p><status>: status of the device.</p>
	<p><message>: description of the task.</p>
	<p><history>:</p>
	<p><name>:</p>
	<p><percentage>:</p>
	<p><description>:</p>

Example response:

```

<ns3:getTaskListResponse>
  <errorMsg>
    <errorCode>0</errorCode>
    <errorMsg>Get task ID detail successfully</errorMsg>
  </errorMsg>
  <taskList>
    <taskId>1</taskId>
    <source>5</source>
    <description>system checkpoint task</description>
    <userID>admin</userID>
    <status>4</status>
    <startTime>2009-09-29T15:18:22Z</startTime>
    <deviceList>
      <devName>create system checkpoint</devName>
      <ip>0.0.0.0</ip>
      <status>4</status>
      <message>Create system checkpoint succeed</message>
      <history>
        <name>create system checkpoint</name>
        <percentage>0</percentage>
        <description>task start ...</description>
      </history>
      <history>
        <name>create system checkpoint</name>
    
```

```

        <percentage>5</percentage>
        <description>Lock system succeed</description>
    </history>
    ...
    ...
</deviceList>
</taskList>
</ns3:getTaskListResponse>
    
```

getTCLRootFile Use this request to get information about the TCLRoot file as defined on the FortiManager unit. Only an administrator with the Super_User profile can run this command.

Table 30: getTCLRoot request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin

Example request:

```

<r20:getFileRequest>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <fileName></fileName>
  <fileOffset>1</fileOffset>
  <fileMaxLen>10</fileMaxLen>
  <fileEncode>0</fileEncode>
</r20:getFileRequest>
    
```

The response is a series of <return> tags, each containing information about the file.

Table 31: getFileRequest response fields

<fileName>	Shows the name of the file. Note: the file name cannot start with a or a ~ character.
<fileOffset>	Indicates the starting point in the receiving file. This must be a positive number and must be smaller than the file.
<fileMaxLen>	Indicates the maximum size of the received file. Must be a positive number.
<fileEncode>	Indicates the encoding method of the receiving file: 0 - base64 1 - hexadecimal base 2 -raw data

Example response

If the task is successful, you will get a message stating that. If the task is not successful, for example a file called root does not exist, you will get a message similar to the one below:

```

:<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
    
```

```
<faultcode>SOAP-ENV:Client</faultcode>
<faultstring>File does not exist</faultstring>
<detail>root</detail>
</SOAP-ENV:Fault>
</SOAP-ENV:Body>
```

listRevisionId Use this request to get a list of the revisions as defined on the FortiManager unit. Only an administrator with the Super_User profile can run this command.

Table 32: getAdomList request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<devId>	Enter the device id number.
<serialNumber>	Enter the device serial number.
<revName>	Enter the name of the revision file.

Example request:

```
<r20:listRevisionId>
  <servicePass>
    <password>?</password>
    <userID>admin</userID>
  </servicePass>
  <devId>?</devId>
  <serialNumber>?</serialNumber>
  <revName>?</revName>
</r20:listRevisionId>
```

The response is a series of <return> tags, each containing information about the revisions.

Table 33: listRevisionId response fields

<errorMsg>	Shows the error message.
<devId>	Indicates the device id number.

Example response:

If the task is successful, you will get the revision information. If the task is unsuccessful, you will get an error message.

retrieveConfig Use this request to get a copy of the configuration as defined on the FortiManager unit. Only an administrator with the Super_User profile can run this command.

Table 34: retrieveConfig request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin

Table 34: retrieveConfig request fields

<devid>	Enter the device id or the device group id number.
<serialNumber>	Enter the device serial number.
<newRevName>	Enter the new name of the revision file. The length should be from 1 to 49 characters.

Example request:

```
<r20:retrieveConfig>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
</r20:retrieveConfig>
```

The response is a series of <return> tags, each containing information about the configuration.

Table 35: retrieveConfig response fields

<errorMsg>	Shows the error message.
<taskId>	Indicates the task id, if there are more than 2 devices.

Example response:

```
<ns3:retrieveConfigResponse>
  <errorMsg>
    <errorCode>-104</errorCode>
    <errorMsg>run retrieveConfig task failed</errorMsg>
  </errorMsg>
</ns3:retrieveConfigResponse>
```

revertConfig Use this request to revert to the previous configuration on the FortiManager unit. Only an administrator with the Super_User profile can run this command.

Table 36: revertConfig request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<devid>	Enter the device id or the device group id number.
<serialNumber>	Enter the device serial number.
<revId>	Enter the revision id number.

Example request:

```
<r20:revertConfig>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
</r20:revertConfig>
```

The response indicates if the configuration reverted successfully or not.

Table 37: revertConfig response fields

<errorMsg>	Indicates the message if an error occurred.
-------------------------	---

Example response:

```
<r20:revertConfig>
  <errorMsg>
    <errorCode>104 </errorCode>
    errorMsg>Revert revision 1 on deviceId 661
      successful</errorMs
  </r20:revertConfig>
  </errorMsg>
</r20:revertConfig>
```

Working with scripts

You can upload scripts to the FortiManager unit and execute them on the FortiManager database or on a managed device. If your scripts make configuration changes to the database, you can install the changes onto the affected devices.

createScript Use this request to upload a script to the FortiManager unit.

Table 38: createScript request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<name>	Enter the script name.
<description>	Optional brief description
<content>	The script.
<overwrite>	1 to overwrite an existing script of that name, otherwise, 0.

Example request:

```
<r20:createScript>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <name>xml_script1</name>
  <description>Generated by XML API</description>
  <content>
config firewall address
edit "33"
  set subnet 33.33.33.33 255.255.255.0
end
  </content>
  <overwrite>1</overwrite>
</r20:createScript>
```

The response is a `<return>` value of 0 if successful, 1 if not. If `<overwrite>` was 0, `createScript` can fail because there is already a script of that name on the FortiManager unit.

Example response: script created

```
<ns3:createScriptResponse>
  <return>0</return>
</ns3:createScriptResponse>
```

deleteScript Use this request to delete a script from the FortiManager unit.

Table 39: deleteScript request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<name>	The name of the script to delete.

Example request:

```
<r20:deleteScript>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <name>xml_script1</name>
</r20:deleteScript>
```

If the script is found and deleted, the response is empty. If the script could not be found, Web Services returns an error message.

Response example: script was deleted

```
<ns3:deleteScriptResponse/>
```

Example response: script not found

```
<faultcode>SOAP-ENV:Client</faultcode>
<faultstring>script xml_script1 is not found</faultstring>
<detail>
  <error xmlns="http://localhost/">script xml_script1 is not
    found</error>
</detail>
```

getScript Use this command to retrieve a script from the FortiManager unit. This is a way to verify the contents of the script. Also, you could modify the script and use the createScript request to update the script on the FortiManager unit.

Table 40: getScript request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<name>	The script name.

Example request:

```
<r20:getScript>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <name>script1</name>
</r20:getScript>
```

The response is a return tag that includes the script content and information about the script.

Table 41: getScript response fields

<content>	Script content
<description>	Administrator password
<name>	Script name
<oid>	Object ID for script

Example response

```
<ns3:getScriptResponse>
  <return>
    <content>
      ... script ...
    </content>
    <description>Generated by XML API</description>
    <name>script1</name>
    <oid>14</oid>
  </return>
</ns3:getScriptResponse>
```

getScriptLog Use this request to get a log of a script from the FortiManager unit.

Table 42: deleteScript request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<devId>	Enter the id number of the device.
<serialNumber>	Enter the serial number of the device.
<logId>	Enter the id number of the log.

Example request:

```
<r20:getScriptLog>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <devId>4755</devId>
  <serialNumber>FG5A253E06500028</serialNumber>
  <logId>log1</logId>
</r20:getScriptLog>
```

Example response

If the task is successful, you will get the log, if not, you will get an error message.

getScriptLogSummary Use this request to get a summary of a script log from the FortiManager unit.

Table 43: deleteScript request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<devId>	Enter the id number of the device.
<serialNumber>	Enter the serial number of the device.
<maxLogs>	Enter the id number of the log.

Example request:

```
<r20:getScriptLogSummary>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <devId>4755</devId>
  <serialNumber>FG5A253E06500028</serialNumber>
  <maxLogs>2</maxLogs>
</r20:getScriptLogSummary>
```

Example of the response:

If the task is successful, you will get a message stating that the summary was created successfully.

If the task is unsuccessful, you will get a message similar to the one below. The details will vary, depending on the error.

```
<faultcode>SOAP-ENV:Client</faultcode>
  <faultstring>No script log</faultstring>
  <detail>
    <error xmlns="http://localhost/">No script log</error>
  </detail>
```

installConfig When you have made configuration changes on the global or device database with your scripts, use this request to install the changes to the devices.

Table 44: installConfig request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<devId>	The Device ID. This is the primary device identifier. You can omit this field and use <serialNumber> instead.
<serialNumber>	Serial number of the device. This device identifier is secondary to <devId>.
<pkgoid>	Enter the package object identifier for a policy package.

Example request:

```
<r20:installConfig>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <devId>109</devId>
  <serialNumber>FGT1002801021024</serialNumber>
</r20:installConfig>
```

If the installation is successful, the response is empty. Otherwise, Web Services returns an error message.

Example response - updated configuration installed successfully:

```
<ns3:installConfigResponse/>
```

Example response - updated configuration could not be installed:

```
<faultcode>SOAP-ENV:Client</faultcode>
<faultstring>Run install+save on deviceId 109
failed</faultstring>
<detail>
  <error xmlns="http://localhost/">Run install+save on deviceId
    109 failed</error>
</detail>
```

runScript Use this request to run a script. You can run a script

- on the global database
- on the device database
- on the managed device

Table 45: runScript request fields

<password>	Administrator password
<userID>	Administrator ID, e.g. admin
<name>	The name of the script to run.
<devId>	Provide the Device ID when you run a script on the device or device database. You can also omit the <devId> field and use <serialNumber> to identify the unit. Set <devId> to -1 when you run the script on the global database.
<serialNumber>	Serial number of the device. This device identifier is secondary to <devId>.
<runOnDB>	1 – run on global or device database, depending on <devId>. 0 – run on the device. Specify <devId> or <serialNumber>.
<pkgoid>	Enter the package object identifier for a policy package.

Example request:

```
<r20:runScript>
  <servicePass>
    <password></password>
    <userID>admin</userID>
  </servicePass>
  <name>xml_script1</name>
  <devId>109</devId>
  <serialNumber>FGT1002801021024</serialNumber>
  <runOnDB>1</runOnDB>
</r20:runScript>
```

If the script runs successfully, the response is empty. Otherwise, Web Services returns an error message.

Example response - script ran successfully:

```
<ns3:runScriptResponse/>
```



A

- addAdom, 9
- addDevice, 10
- addGroup, 11

C

- createScript, 30

D

- deleteAdom, 11
- deleteConfigRev, 12
- deleteDevice, 13
- deleteGroup, 13
- deleteScript, 31
- deleting scripts, 31
- device IDs
 - obtaining, 18

E

- editAdomMembership, 14
- editGroupMembership, 14
- executing scripts, 34

G

- getAdomList, 15
- getConfig, 19
- getConfigRevisionHistory, 21
- getDeviceLicenseList, 22
- getDeviceList, 16
- getDevices, 18
- getGroupList, 22
- getPackageList, 23
- getScript, 31
- getScriptLog, 32
- getScriptLogSummary, 33
- getTaskList, 24
- getTCLRootFile, 26

I

- installConfig, 33
- installing configuration changes, 33

L

- listing
 - ADOMs, 15
 - device configuration, 19
 - device configuration history, 21
 - device groups, 22
 - device information, 18
 - devices, 16
- listRevisionId, 27

R

- retrieveConfig, 27
- retrieving scripts, 31
- revertConfig, 28
- running scripts, 34
- runScript, 34

S

- scripts
 - deleting, 31
 - executing, 34
 - listing, 31
 - retrieving, 31
 - uploading, 30

U

- uploading scripts, 30

W

- Web Services
 - enabling, 8
- WSDL file
 - obtaining, 8

F **ORTINET**®

