Zero Trust for Linux Admins

With Open-Source IAM

Thomas Cameron, RHCA, AWS SA Pro (he/him)
Senior Principal Solution Architect, Automation
thomas@redhat.com

Red Hat

What we'll discuss today

- Introductions: Who Am I?
- Introduction: De-fuzzing the Buzzword
- The Backbone: Centralized Identity with FreeIPA
- Policy-Based Access: HBAC and sudo

Source:
Insert source data here
Insert source data here

What we'll discuss today

- Modern Auth: SSH Certificate Authorities
- Local Enforcement: Firewalld & SELinux
- Automation & Lessons Learned
- Conclusion & Q&A

Source:
Insert source data here
Insert source data here

Introductions:

Who Am I?

I’m Thomas Cameron!

Most importantly, I’ve been married to my partner Theresa since September 6th, 1997. We have two kids, a daughter (Nicole) who is an EMT, and a son (Simon) who is in nursing school.

- I believe in the ideal that this country should be a place where everyone has the freedom to be themselves and pursue happiness without fear.
- I’m a firm believer in the founding principle that we are **all** created equal and deserve the same dignity and rights.

Source:
Insert source data here
Insert source data here

I’m Thomas Cameron!

- My focus is on creating a community which is welcoming and equitable for every person.
- I support a vision of America where opportunity and safety are available to all, regardless of who they are, where they’re from, or who they love.
- I value fairness, and for me, that means standing up for equal rights and representation for everyone.

Source:
Insert source data here
Insert source data here

I’m Thomas Cameron!

- If this is a problem, I’ll be sorry to see you go.

Source:
Insert source data here
Insert source data here

I’m Thomas Cameron!

- Professionally:
- I’ve been in IT since 1993.
- I started out with Novell NetWare
- Worked with Microsoft technologies
- Discovered Linux in 1995. I’ve been an Open Source geek ever since.
- I’ve managed large scale Linux environments for organizations from banking to real estate to chip manufacturing.
- I worked at Red Hat from 2005 through 2019, as a chief architect and global technical evangelist.
- I then went to AWS, where I was a senior technical trainer.
- In September of ‘24, I came back to Red Hat as a senior principal solution architect specializing in Ansible Automation.

Source:
Insert source data here
Insert source data here

De-fuzzing the Buzzword

Melted Perimeter

Melted Perimeter

It used to be that folks would come into the office, log in to their local network, and do their work. We had security at the physical level, and were protected by firewalls.

If folks needed remote access, they had to log in via a VPN.

Source:
Insert source data here
Insert source data here

Melted Perimeter

Now, we have SaaS for everything, and folks work remotely. We are expected to have work email and Slack/Teams/GChat/whatever installed on our phones (we're not going to argue the merits of this today - I only have an hour).

Source:
Insert source data here
Insert source data here

Credit Gemini

Melted Perimeter

The Flaw: Once a packet or a user crossed the drawbridge, they had "Lateral Liberty." We assumed that because you were on the internal VLAN, you were supposed to be there.

The concept of “internal vs. external” is DEAD. ☠️

Source:
Insert source data here
Insert source data here

The "Melted" Reality: Assume Breach

In a Zero Trust architecture, we treat the internal network as if it were a public Starbucks Wi-Fi.

- Network Location does not equal Trust: Just because an IP address comes from your local subnet doesn't mean it gets a pass.
- The Identity is the New Perimeter: Instead of a IP-based firewall rule, the "wall" is now a Kerberos ticket or an SSH Certificate tied to a specific human or service account.

Source:
Insert source data here
Insert source data here

The "Melted" Reality: Explicit Verification

In a Zero Trust architecture, **identity** is superior to IP address

- Even if you're logged into a VPN, do you **really** have permission to do what you're asking to?

Source:
Insert source data here
Insert source data here

The "Melted" Reality: Least Privilege

In a Zero Trust architecture, we grant the minimum necessary privileges to perform tasks

- Now we know who you are, what do you have permission to do?
- What apps can you use?
- What hosts can you log into?

Source:
Insert source data here
Insert source data here

The Linux Connection

In Fedora/CentOS Stream/RHEL and others, we aren't just hardening the firewall to keep people out; we are using SSSD and FreeIPA to ensure that even if someone is 'inside' the network, they can't see or touch a single byte of data without a cryptographically proven identity.

Source:
Insert source data here
Insert source data here

WARNING: SALES CONTENT

Steel yourselves!

The Linux Connection

Everything we're going to discuss today is available in community supported distros like Fedora and CentOS Stream, and in commercially supported Red Hat Enterprise Linux.

Source:
Insert source data here
Insert source data here

The Backbone

Centralized Identity with FreeIPA

The Problem

Local /etc/passwd and hand-copied SSH keys are a security **nightmare**.

Source:
Insert source data here
Insert source data here

The Solution

Red Hat Identity Manager/FreeIPA as the "Linux Domain Controller."

Source:
Insert source data here
Insert source data here

DEMO: Setting up IdM

DEMO: Registering to the Directory Server

DEMO: Show Registered Servers

Policy-Based Access: HBAC and Sudo

Host-Based Access Control (HBAC)

Moving beyond "can you log in" to "where are you allowed to be?"

- Example: DBAs can touch DB servers, but Web Devs cannot.

Source:
Insert source data here
Insert source data here

Centralized Sudo

Stop editing /etc/sudoers on each host!

- Define a rule in FreeIPA.
- It propagates to the whole fleet via SSSD.

Source:
Insert source data here
Insert source data here

DEMO: Managing Access
Policy/HBAC

DEMO: Managing sudo

Policy/sudo

Modern Auth: SSH Certificate Authorities

SSH Certificate Authorities

The Risk

- Static SSH keys are "forever credentials." If a laptop is stolen, you're rotating 1,000 keys.

Source:
Insert source data here
Insert source data here

SSH Certificate Authorities

The Pivot: Short-lived SSH Certificates

- Users authenticate to a CA (like Vault or FreeIPA's KDB) and get a 1-hour certificate

Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

This architecture uses the FreeIPA Kerberos Database (KDB) as the source of truth for host identities, providing "Zero-Touch" SSH trust across the environment.

1. Prerequisites
    a. IdM Server: Red Hat IdM / FreeIPA installed with CA.
    b. Nodes: RHEL/Fedora/CentOS Stream joined to the realm via ipa-client-install.
    c. SELinux: Enabled (Enforcing).

Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

This architecture uses the FreeIPA Kerberos Database (KDB) as the source of truth for host identities, providing "Zero-Touch" SSH trust across the environment.

1. Server-Side Configuration (IDM Master)
   a. To allow hosts to manage their own certificates in the KDB, ensure they have the proper permissions.

```
# Grant the host permission to manage its own host record

# Replace 'workstation1.redhat.lan' with your node's FQDN

ipa host-add-managedby workstation1.redhat.lan
--hosts=workstation1.redhat.lan
```


Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

Node Configuration (The "Target" Server)

- Run these steps on every node deployed via ZTP to enable automated rotation.

1. Certmonger requires the classic PEM format to read and manage the keys.

```
# 1. Convert OpenSSH format to PEM (PKCS#1)

ssh-keygen -p -N "" -m PEM -f /etc/ssh/ssh_host_rsa_key

# 2. Fix SELinux contexts for the SSH directory

restorecon -v /etc/ssh/ssh_host_rsa_key
```


Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

Deploy Custom SELinux Policy

- Run these steps on every node deployed via ZTP to enable automated rotation.

1. Create a module to allow certmonger to manage files in /etc/ssh.

Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

Deploy Custom SELinux Policy

```
# Create certmongerlocal.te

cat <<EOF > certmongerlocal.te

module certmongerlocal 1.0;

require {

    type certmonger_t;

    type sshd_key_t;

    class file { getattr open read write };

}

allow certmonger_t sshd_key_t:file { getattr open read write };

EOF
```


Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

Deploy Custom SELinux Policy

```
# Compile and install

checkmodule -M -m -o certmongerlocal.mod certmongerlocal.te

semodule_package -o certmongerlocal.pp -m certmongerlocal.mod

semodule -i certmongerlocal.pp
```


Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

Enroll in KDB for Rotation

- Use ipa-getcert to request the certificate. Certmonger will now monitor this file and auto-renew it 30 days before expiration.

```
# Request the cert using the Host Principal

ipa-getcert request \

    -K "host/$(hostname -f)@REDHAT.LAN" \

    -f /etc/ssh/ssh_host_rsa_key-cert.pub \

    -k /etc/ssh/ssh_host_rsa_key \

    -C "/usr/bin/systemctl reload sshd"
```


Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

Update SSH Daemon

- Add HostCertificate to the file:

```
echo "HostCertificate /etc/ssh/ssh_host_rsa_key-cert.pub" >>
/etc/ssh/sshd_config

systemctl reload sshd
```


Source:
Insert source data here
Insert source data here

Automated SSH Key Rotation & Transparent Trust (FreeIPA)

Client-Side Configuration (The "Admin" Machine)

- To achieve Transparent Trust (no "Unknown Host" prompts), configure the SSH client to trust the KDB via SSSD.
- Edit /etc/ssh/ssh_config.d/99-ipa-trust.conf:

```
# Request the cert using the Host Principal

Host *.redhat.lan

    GlobalKnownHostsFile /var/lib/sss/pubconf/known_hosts

    ProxyCommand /usr/bin/sss_ssh_knownhostsproxy -p %p %h
```


Source:
Insert source data here
Insert source data here

(Time permitting)

DEMO: Hashicorp Vault

As A CA

(Time permitting)

DEMO: IdM

As a CA

Local Enforcement: Firewalld & SELinux

Local Enforcement: Firewalld & SELinux

Micro-segmentation

- Using firewalld zones to ensure only the IPA server can talk to the Kerberos ports.

Source:
Insert source data here
Insert source data here

Local Enforcement: Firewalld & SELinux

SELinux as the Last Line

- Why "Assume Breach" means keeping SELinux in Enforcing mode to contain a compromised service.

Source:
Insert source data here
Insert source data here

Automation & Lessons Learned

Automation & Lessons Learned

GitOps for Identity

- Using Ansible Core to manage IPA users and HBAC rules as code.

Source:
Insert source data here
Insert source data here

Automation & Lessons Learned

Pitfalls to Avoid

- Time Sync: Kerberos will break if NTP drifts by >5 minutes.
- Break-glass Accounts: Always keep one local admin user in case the network/IPA is down.
- DNS: FreeIPA is highly dependent on clean SRV records.

Source:
Insert source data here
Insert source data here

Conclusion & Q&A

Conclusion & Q&A

Summary

- Zero Trust isn't a product you buy; it's a configuration state you maintain.

Source:
Insert source data here
Insert source data here

Conclusion & Q&A

Call to Action

- Start by migrating one group’s sudo rules to FreeIPA this week.

Source:
Insert source data here
Insert source data here

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

facebook.com/RedHat

youtube.com/@redhat

x.com/RedHat

Red Hat