WHITE PAPER

# The Perfect SSH Storm

## The disruptive effect of SSH threats and how to prepare, prevent and stop them

Venafi

# Introduction

IT and security teams increasingly are using Secure Shell (SSH) to safeguard administrative access and automated processes for their organizations. As a result of this continuous IT workload redesign and an upsurge in the number of machines, more is being required of SSH connections. Because of a lack of oversight, however, SSH keys are spawning numerous vulnerabilities that may be easily exploited and make recovery difficult. This perfect storm of exploding growth and lack of management creates a high security risk that has hit many organizations without warning.

Just as usernames and passwords authenticate people on networks, machines must be authenticated using machine identities to secure connections. Different types of machine identities include SSH keys, SSL/TLS keys and certificates, and endpoint, user and code signing certificates—all of which must be managed and secured. Although SSH keys are a vital machine identity type used to identify and authenticate administrators and machines for critical business functions, they are routinely left untracked, unmanaged and unmonitored.

This paper provides a brief introduction on SSH and its popularity, SSH keys as machine identities, and seven common SSH machine identity threat risks based on real-world examples. Also, IT and security risk managers will find an essential overview of methods and steps required to support businesses with sustainable growth and management of their SSH machine identities.

# Growing Dependence on Secure Shell

Built into most operating and network systems, SSH has become the de facto interface standard for remote system access. SSH popularity is due largely to its security features, versatile usage and baked-in automation. Today, this broadly adopted cryptographic protocol is in use by a majority of system admins as well as many automated processes.

When a system admin executes an SSH command, the SSH client and server engage in six steps:

1. Sending a connection request;
2. Authenticating back to the client using the unique server host SSH key;
3. Setting up an encrypted channel;
4. Authorizing session access by using passwords or authorized keys;
5. Submitting a session info request by executing shell commands (i.e., dir); and
6. Returning session info when the SSH server sends the command return data back to the client (i.e., a list of files).

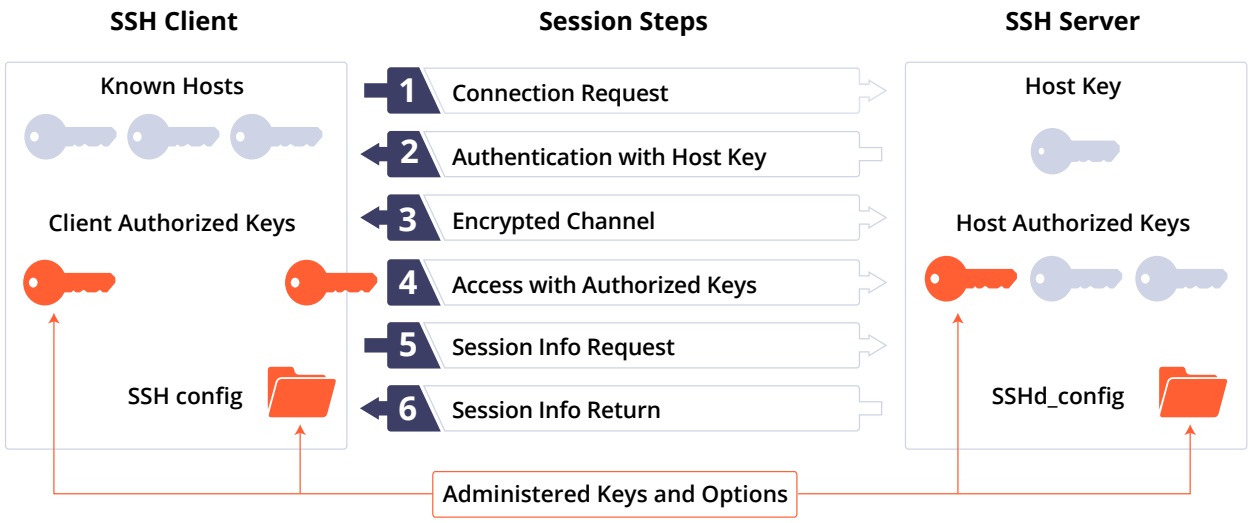


**Figure 1: SSH Connection Overview—One SSH Command Equals Six Steps**

# The Role of SSH Keys

During the connection build process, SSH leverages two types of SSH keys as machine identities. First, SSH uses a **host key** to guarantee the authenticity of the server and create the encrypted tunnel. Secondly, SSH users can place **authorized keys** on the server to grant them access without using passwords, thus simplifying day-to-day work. Both host and authorized keys work in public-private pairs and must be administered together with a slew of security options (config files) such as the encryption algorithm, access levels, port forwarding, key length and passphrase. (See Figure 1: SSH Connection Overview.)

The net result of these SSH machine identities is a strong mesh of automated, secure, highly trusted paths inside or from outside-to-inside enterprise business critical assets. Consequently, these SSH identities and the vital connections they provide are common across most cloud and on-premises environments.

# 7 SSH Threat Risks to the Enterprise

Because of its prevalence and adaptability to multiple environments, SSH keys and the connections they enable have gained in popularity significantly over the last several years. Yet, SSH deployment and its related configuration can leave organizations vulnerable if not done securely. Here is a list of the most imminent threat risks:

1. **Key sprawl:** SSH has many moving parts that can be touched by its users or the system admins managing the SSH keys that serve as machine identities. A lack of governance in creation and management of the various SSH keys can lead to the reckless proliferation of keys which can, in turn, lead to unauthorized access that is difficult to detect. For instance, keys delivering privileged access can get duplicated or shared between users, making the connections less private and more prone to attacks.
2. **Missing controls:** As business-critical SSH connections expand uncontrolled, oversight of the SSH keys, owners, access level and authorized assets often get lost, resulting in a chaotic mesh of trusted connections. A common risk discovered by Venafi during risk assessments is the abundance of unnecessary SSH root keys that violate data privacy policies and generate unwanted exposure.
3. **No SSH key expiration:** SSH keys never expire, which means that when a system admin leaves the organization or an IT automation process gets removed, related keys may still be located in various files and accessed by unauthorized users.
4. **Lost or stolen SSH keys:** Like many system elements, SSH keys are defined in a file, easy to recognize and stored on both sides of a connection. As a result, malware or insider threats could lead to key theft, opening the door for an intruder to start a privileged SSH administrative session.
5. **Lateral movements and pivoting:** Once a system has been compromised, adversaries like to move around and expand their access. A dense and uncontrolled environment of SSH key-enabled connections can form a vast web for adversaries to pivot from asset to asset, using keys found in various user accounts.
6. **Obscured and exfiltrated data:** Adversaries like to "live off the land" or, simply said, attackers like to hide within the infrastructure using readily available tools, like the SSH protocol, to redirect and exfiltrate data without being detected by traditional controls. SSH enables traffic redirects and allows its users to set up a listening port on a client and tunnel data through an encrypted channel to an exit server port or vice versa. As a result, encrypted SSH connections can also be abused by attackers to exfiltrate data without being detected.
7. **Slow incident response processes:** When an incident occurs, responders need to take action and remove all potential access paths available to the intruder. A dense and uncontrolled mesh of SSH machine identities with hundreds of thousands—if not millions—of keys can be hard to clean up, consuming a lot of resources and allowing adversaries extra time to leverage the privileged access they have acquired.

While these threats are mostly a result of poorly managed SSH keys, when combined, they form a sequence that expands the threat, leading to the perfect SSH storm. (Figure 2 below shows how these threats build.) This leaves business-critical assets unprotected and defenders unarmed.

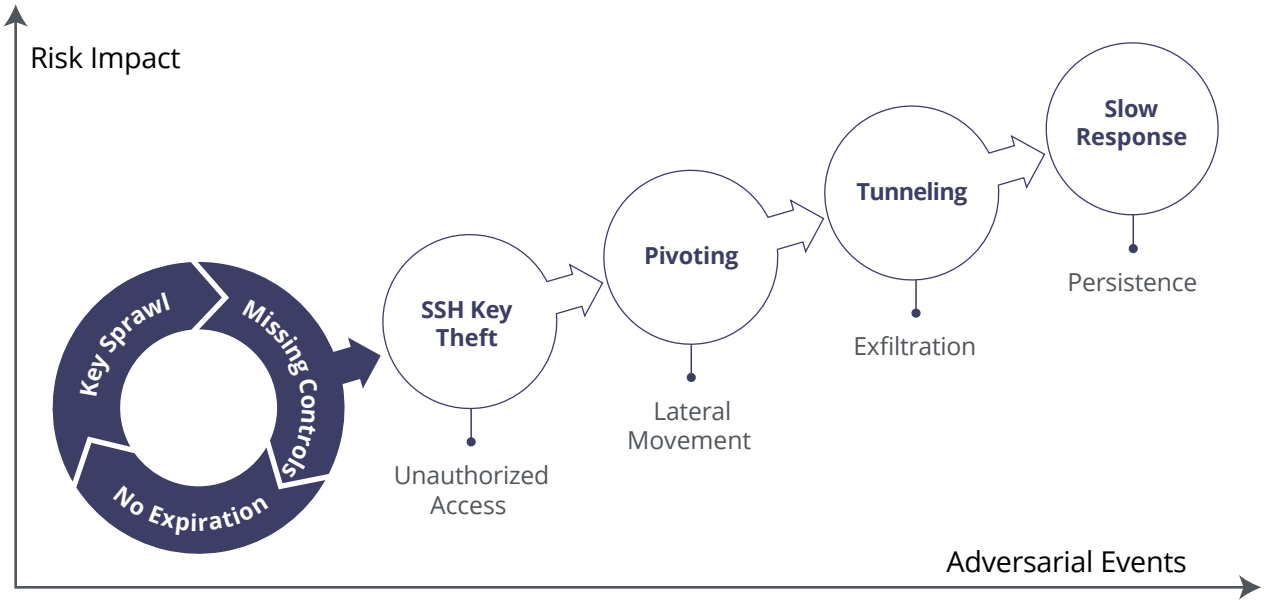


**Figure 2: SSH Threat Risks and How They Build to a Perfect Storm**

# SSH Stories from the Trenches: From Bad to Worse

It did not take long for adversaries to discover these SSH security vulnerabilities. Over the past decade, a number of SSH exploits, threat campaigns and other security incident horror stories have made headlines. The following are some extracts:

- **No SSH key visibility:** According to a 2017 survey conducted by Dimension Research, only 10 percent of customers have a complete and accurate SSH key inventory, which leaves them blind to SSH key sprawl and their level of vulnerability.[1]
- **Golang-based cryptomining campaign:** In July 2019, F5 threat researchers uncovered a cryptomining campaign that delivered Golang-based (i.e., based on Google Go language) malware targeting Linux servers. One of the malware propagation methods used SSH credential enumeration and lateral movement, exploiting uncovered SSH keys, which led to several thousand infected hosts.[2]
- **SSH vulnerabilities:** In May 2019, Cisco published a critical vulnerability with a 9.8 out of 10 Common Vulnerability Scoring System (CVSS) score that leveraged never expiring high-privilege SSH key pairs left on shipped Cisco devices. Attackers could exploit this vulnerability by opening an SSH connection via IPv6 to a targeted device using the extracted key materials.[3]
- **Public keys in the wild:** In wake of the 2014 Sony Pictures breach, the cracking group "Guardians of Peace" started to leak stolen corporate data. At least 20 SSH keys were allegedly discovered by inquisitive users downloading the released data, with one SSH key named AkamaiPrivateKey.ppk.[4]

# Management and Security with Visibility, Intelligence and Automation

Getting control over a chaotic collection of SSH keys is not as unchartered as it sounds. Organizations, including SANS and the National Institute of Standards and Technology (NIST), have highlighted the problem and published guidelines like NIST-IR 7966.[5] This last report especially emphasizes the need for more structural SSH key lifecycle practices, monitoring, inventory and automation, which can be summarized in three essential SSH management characteristics:

**Visibility:** As most organizations have no insights into the number of SSH keys in use, visibility is, in most cases, the starting point for improving SSH key management. By running solid discovery, creating an inventory and mapping SSH keys pairs, InfoSec teams can get a clear overview of all SSH keys and trusted relationships, including users, hosts and configuration options. A mix of discovery mechanisms and flexible reporting capabilities will help make this task run as quickly and smoothly as needed to find keys across the enterprise.

**Intelligence:** Once an inventory of keys, access rights and related hosts are known, administrators need to apply intelligence to find high-risk connections. A lack of visibility into orphaned, shared, weak or root keys can lead to unauthorized access and must be immediately reported for analytic review. Enterprises must also be able to create their own rules to identify out-of-policy practices like cross-environment key usage, improper key lengths or aged keys. Monitoring these generally accepted SSH risk audit practices as well as enterprise-specific policies for actionable alerts is, of course, essential.

**Automation:** Time and resources are precious commodities for InfoSec and security operations teams. Automated capabilities like "single-click" machine-assisted key rotation, scheduled bulk cleanup of out-of-policy keys or self-service managed key generation for system admins should be put into place to improve efficiencies, tighten security and reduce errors introduced by manual processes.

# 4 Steps to Secure SSH Usage

Venafi SSH Protect applies these capabilities to help enterprises manage SSH across thousands of hosts with millions of keys in use. After enterprises work with Venafi through an initial risk assessment, the following four steps will secure the organization's SSH usage:

**Step 1: Build a comprehensive inventory.** Venafi enables a complete and accurate inventory, which is required for SSH visibility and control across the network. An SSH inventory should include information on all SSH servers, private keys (the authorized keys that grant SSH access), connections and any SSH configurations that limit access.

**Step 2: Identify vulnerabilities.** To reduce the risk of breach and compromise, Venafi helps organizations analyze their SSH key inventory to identify which SSH keys and servers are vulnerable and why.

**Step 3: Remediate.** Once vulnerabilities are identified, they must be fixed quickly to prevent ongoing exposure to a potential breach. Venafi delivers automated responses to SSH issues, including removal of unauthorized keys, rotation/replacement of weak and old keys, removal of SSH root access, removal of duplicate private keys and enforcement of security controls that limit the accessibility and use of SSH keys.

**Step 4: Monitor.** To meet ongoing security and compliance requirements, Venafi delivers continuous, automated monitoring and tracking of SSH keys. Venafi also supports SSH audit practices that regularly review SSH entitlements, assess risk, avoid compliance violations and increase accountability for identity and access management.

# Benefits

After applying these steps, IT and security risk organizations typically achieve the following:

- Reduce threat risk by removing unnecessary high-risk keys such as unused and shared high-privilege SSH keys.
- Conduct security audits on time and within budget, eliminating potential fines related to improper SSH access usage.
- Improve SSH rotation efforts, leading to faster response, shortened exploitation windows from vulnerable SSH keys and lower overall risk exposure.
- Coordinate the full SSH key management lifecycle and deploy SSH identities with InfoSec oversight and control.

# Conclusion

All organizations rely on SSH as an encrypted protocol to authenticate privileged users, establish trusted access and connect administrators and machines. But increased SSH use paired with a lack of proper SSH machine identity management has created a perfect SSH storm that has exposed organizations to SSH key exploitation by adversaries. IT and security risk managers need to apply visibility, intelligence and automation to get full control over the SSH key lifecycle and keep IT business environments safe.

To learn more about Venafi SSH Protect, visit **venafi.com/platform/ssh-protect**.

## References


1. Venafi. 2017 SSH Study Reveals Widespread Lack of Security Controls for SSH Keys. July, 2017.
2. Arghire, Ionut. Securityweek. Cryptomining Campaign Targets Linux Servers with Go Malware. July 5, 2019.
3. Heller, Michael. TechTarget. Cisco SSH Vulnerability Sparks Debate Over Backdoors. May 6, 2019.
4. Paulli, Darren. The Register. Sony Pictures in IT Lock-Down After Alleged Hacker Hosing. November 25, 2014.
5. National Institute of Standards and Technology. NISTIR 7966. Security of Interactive and Automated Access Management Using Secure Shell (SSH). October 2015.