



Orchestration tools: Access risk mitigation & migration to keyless, JIT access

Technical solution brief



Index

Introduction	3
<i>Orchestration tools typically use SSH keys to access targets</i>	3
Mitigating the risks of SSH keys through migration to keyless just-in-time ephemeral access	4
<i>Traditional Ansible access</i>	4
<i>The future of SSH access is keyless</i>	4
Why should you migrate to keyless access with SSH Zero Trust Suite	6

Introduction

Manually performing routine tasks in today's typical enterprise environments eats up time and money and is prone to errors. Thus, organizations with large environments commonly utilize automation orchestration tools, such as Ansible, Puppet, and BMC TrueSight, to simplify software provisioning, configuration management, and application deployment.

Orchestration tools help streamline tasks within large environments, but on the other hand, they can also complicate the environments as they typically use SSH keys as access credentials.

In this document, we will discuss the inherent risks of orchestration tools and how the risks could be solved by migrating from using SSH keys to keyless, just-in-time access.

Orchestration tools typically use SSH keys to access targets

Orchestration tools utilize control nodes to manage routine tasks: A control node manages the target inventory nodes in order to automate standard tasks, such as configuring and provisioning new systems, deploying software, and managing updates. To ensure the security of used communication channels and authenticate the authenticity of change requests, orchestration tools oftentimes utilize SSH keys as the access credentials. In large environments, this equates to the need to provision SSH keys every time a new target is onboarded. Further, the provisioned SSH keys also need to be periodically rotated – that can lead to regularly rotating thousands of keys to satisfy regulatory requirements.

However, SSH keys come with their own risks. They utilize public key cryptography and are more secure access credentials than passwords, but they present significant inherent risks. For example:

- SSH keys provide ease of lateral movement and privilege elevation
- SSH keys can be easily copied
- SSH keys never expire and cannot be centrally revoked

Additionally, managing SSH keys in large environments becomes a burden over time – managing SSH keys in thousands or even millions is time-demanding and resource-heavy.

[Find out more about SSH keys, their risks, and various management approaches in our SSH Key Management Compass here >>>](#)

Mitigating the risks of SSH keys through migration to keyless just-in-time ephemeral access

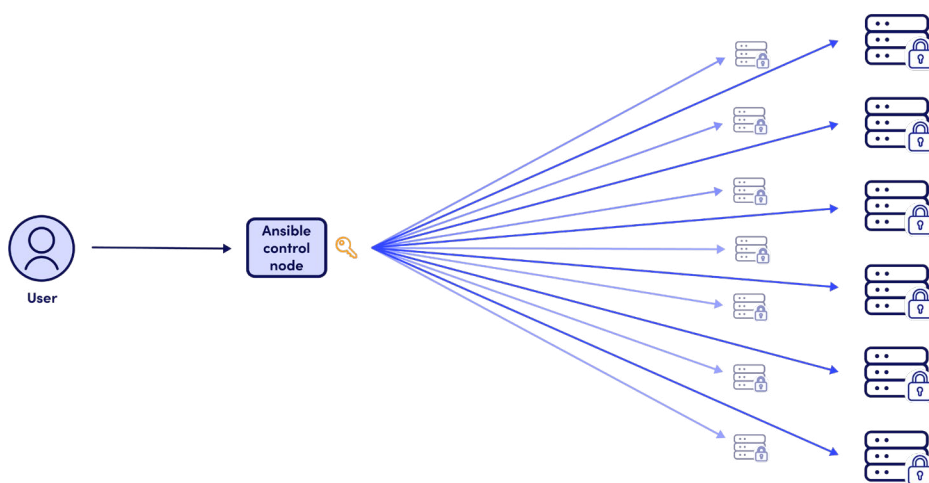
All of the above-mentioned risks and challenges have a solution – the migration from using SSH keys to using keyless just-in-time access enabled with ephemeral, short-lived certificates.

Our SSH Zero Trust solution is an all-encompassing secrets and access management platform that offers a migration path for existing access credentials, which are already in use by automation tools, as well as new access. All done with the efficiency demanded by large IT environments.

Instead of using SSH keys as the default access credentials for automation tools, such as Ansible, going forward you can utilize ephemeral certificates. Below is an example comparison of the traditional access using Ansible and the modern keyless access.

Traditional Ansible access

By default, Ansible uses SSH keys to achieve access to nodes under management. The distribution and configuration of those credentials must be done before Ansible can operate.



A user can access thousands of servers thanks to SSH keys provisioned by an orchestration tool, like Ansible.

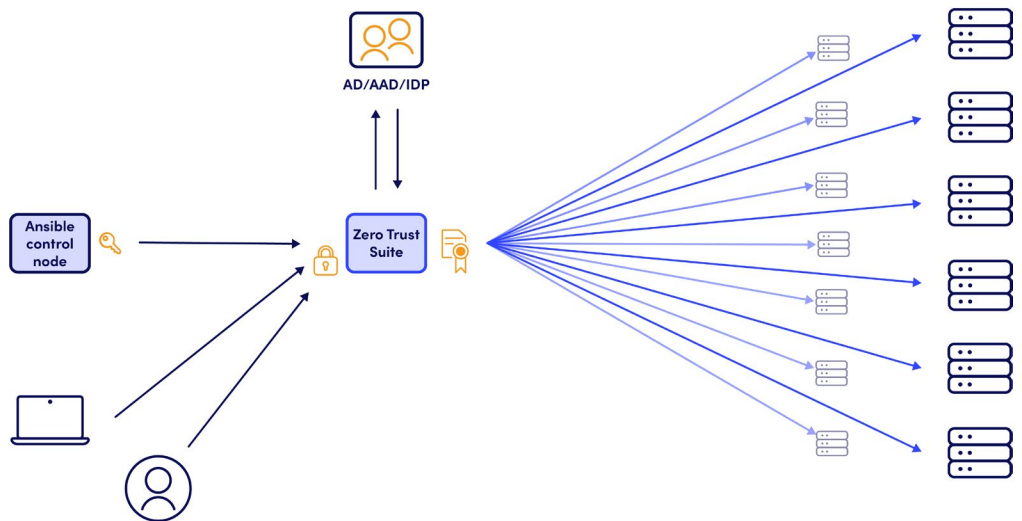
The future of SSH access is keyless

Since SSH keys rely on public key cryptography, they are superior in terms of security compared to other standing credentials, such as passwords.

Additionally, the ease of use is partially the reason why SSH keys are so widespread. However, this widespread usage also exacerbates the inherent disadvantages and increased maintenance needs of SSH keys.

Instead of SSH keys, a viable solution to the typical problems is replacing keys with short-lived certificates which get generated at the time when needed (just-in-time access). Compared to SSH keys:

- Certificate authentication does not require configuration of access credentials on the target nodes.
- Access is granted only after verifying the identity of the user requesting the access.
- The access-granting certificate is generated with a lifespan of minutes, making it unusable if a credential is exported and later compromised.



An example of keyless access utilizing SSH Zero Trust Suite: An orchestration tool, like Ansible, connects to Zero Trust Suite with an SSH key. Users (human as well as machine) connect also to Zero Trust Suite, which verifies their role via integration with AD/AAD/IDP. Zero Trust Suite connects to thousands of servers utilizing short-lived, ephemeral certificates.

From an operational perspective, the migration process from using SSH keys to ephemeral certificates is designed to be fully automated and transparent to the end-user or applications utilizing SSH access. The automation is designed to discover and create an inventory of where SSH key access is used and identify SSH keys and their trust relationships. It automatically performs any necessary configuration to facilitate the migration from standing access credential to the use of ephemeral certificates. And all is clearly visible under a single pane of glass.

The transparency of this process guarantees that no existing infrastructure, scripts, or integrations need to be modified.

Why should you migrate to keyless access with SSH Zero Trust Suite

1. Mitigate risks with role-based access control (RBAC) for SSH access.

TRADITIONAL ACCESS	KEYLESS ACCESS
<p>Accessing nodes using SSH client/server applications does not provide the ability to apply RBAC. Anyone in possession of access credentials can use them until the credentials are explicitly rotated or removed.</p>	<p>With Zero Trust Suite, organizations can automatically apply access rules based on their chosen Single Point of Truth (SPOT), such as an active directory or their preferred identity provider.</p> <p>Zero Trust Suite adjusts instantly to the joiner-leaver-mover processes in highly dynamic environments and ends ongoing connections when the parameters are not satisfied.</p>

2. Reduce configuration overhead significantly by eliminating the need for password and key management on targets. Support the immutable architecture paradigm.

TRADITIONAL ACCESS	KEYLESS ACCESS
<p>Machine-to-machine SSH access often utilizes standing credentials, such as SSH keys and passwords, for the needs of automation, configuration, and monitoring among others.</p> <p>New access credentials need to be explicitly provisioned on target nodes and can be in the hundreds of thousands, especially in the case of granting access for automation tools.</p>	<p>Zero Trust Suite utilizes ephemeral certificates by default and eliminates the need to provision and subsequently rotate SSH keys or passwords on the target servers.</p> <p>The technology behind ephemeral certificate access inherently promotes the immutable architecture paradigm. This access requires only an initial configuration onto the target nodes, which can be performed during the time of deployment, to establish trust with the certificate authority chosen as the credential issuer.</p>

3. Mitigate unauthorized access risks with just-in-time credentials.

TRADITIONAL ACCESS	KEYLESS ACCESS
<p>Traditional risk mitigation for SSH access utilizing standing credentials involves periodic rotation of those credentials, often spanning the entire estate.</p>	<p>Zero Trust Suite eliminates the standing credentials altogether, mitigating the risk associated with compromised SSH keys and passwords.</p>

4. Gain full control over all SSH access.

TRADITIONAL ACCESS	KEYLESS ACCESS
<p>Often, the only SSH access that organizations bring under control is interactive access. However, around 80% of all SSH connections are between machines, such as automation and monitoring.</p>	<p>With Zero Trust Suite, you can achieve controlled access closing and avoid common security issues, such as PAM-bypass.</p>

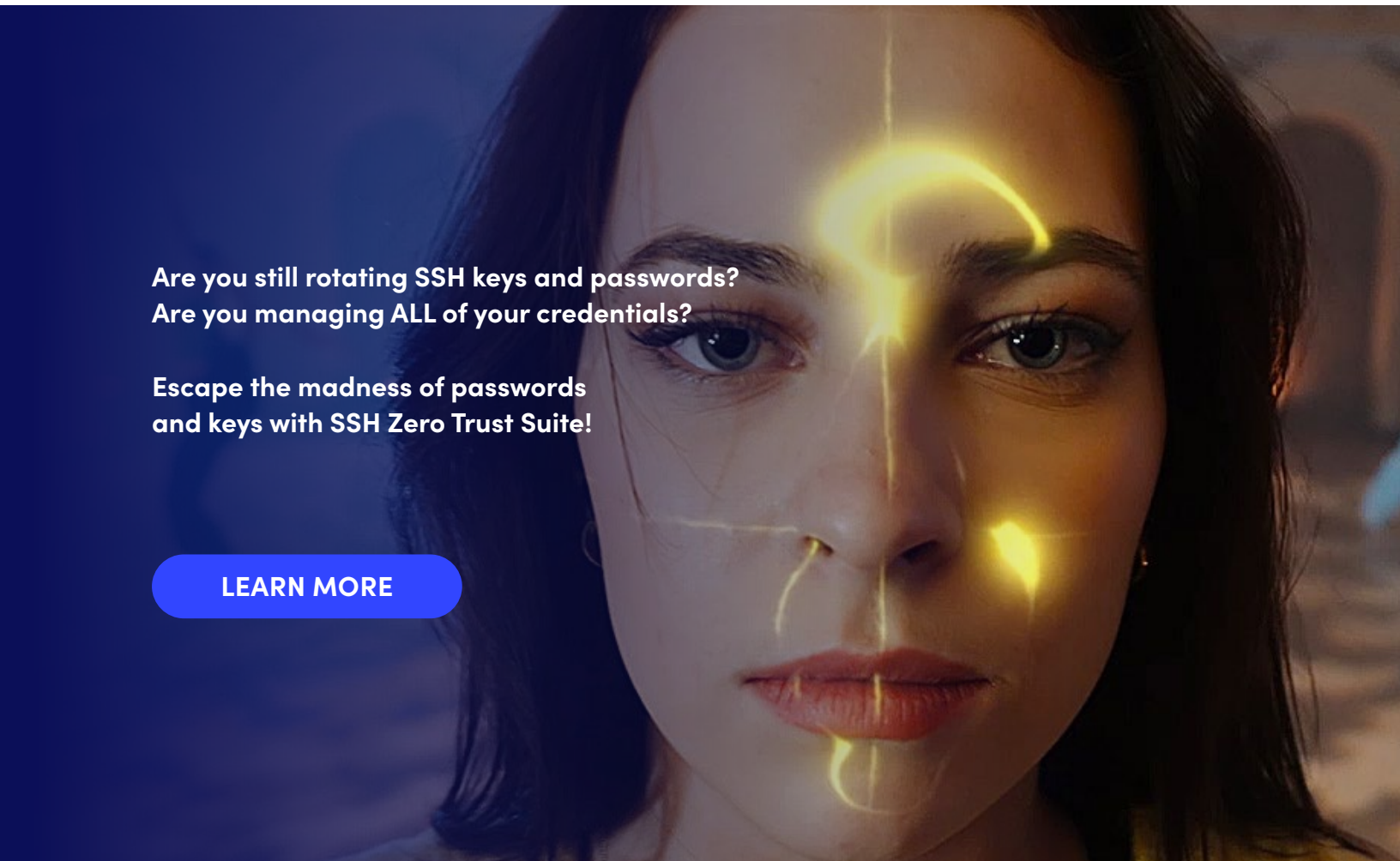


Additionally, the security that SSH connectivity is designed to provide comes paired with total privacy. While security and privacy are desired, they also provide a cloak for adversaries to perform actions on nodes that are opaque to traditional monitoring techniques. Exfiltration of data, establishing connectivity permanence, or unauthorized modification then become trivial and hard to detect.

This alone increases the risk exposure for organizations.

Zero Trust Suite provides a transparent view into the activities initiated by SSH connections. It offers the ability to fully audit and monitor the connections, apply whitelisting rules, and record entire sessions.

It can also apply operational windows and approval flows, utilize the four-eye principle, and even terminate ongoing connections on demand.



**Are you still rotating SSH keys and passwords?
Are you managing ALL of your credentials?**

**Escape the madness of passwords
and keys with SSH Zero Trust Suite!**

LEARN MORE

We'd love to hear from you!

Get in touch with our experts around the world.

GLOBAL HEADQUARTERS

Helsinki

SSH Communications Security
Oyj
Karvaamokuja 2b, Suite 600
FI-00380 Helsinki
Finland
+358 20 500 7000
info.fi@ssh.com

US HEADQUARTERS

New York City

SSH Communications Security
Inc.
66 Hudson Blvd E, Suite 2308
New York, NY, 10001
USA
Tel: +1 (212) 319 3191
info.us@ssh.com

APAC HEADQUARTERS

Singapore

SSH Communications Security
Ltd.
6 Raffles Boulevard, Marina
Square, #03-308
Singapore 039594
Singapore
Tel. +65 6338 7160
sales.asia@ssh.com

Let's get to know each other

Want to find out more about how we safeguard mission-critical access for leading organizations around the world? We'd love to hear from you.

[REQUEST A DEMO](#)