National Cyber Security Centre
a part of GCHQ

# Principles for secure privileged access workstations (PAWs)

How to design and securely build management devices for high-risk system maintenance and administration.

Version 1.0
Published 25 March 2025

# Who is this guidance for?

This guidance helps organisations and cyber security professionals implement effective privileged access workstation (PAW) solutions in organisations. It outlines the essential characteristics of PAWs and practical considerations for implementing them in typical scenarios, in the form of principles.

Where a third party requires high-risk access to your environment, it can also help you assess whether the third party is using appropriately secured devices.

# Why is guidance required?

Digital systems are now integral to the work of any large organisation and to work effectively, authorised staff must be able to configure, operate and maintain them. But the benefits of technology and ease of use must be balanced against the potential for compromise, which can have a significant impact on business, or even on critical national infrastructure (CNI) if the affected organisation is in a CNI sector.

Highly privileged administration interfaces are a prime target for attackers who may want to maliciously access your systems, and threat actors may try to attack the devices used to access these interfaces. Using a well-configured PAW makes this a lot more difficult.

There is no one size fits all when it comes to a PAW, or a single product to cover these controls. Implementing a PAW solution should be approached as a strategy, in line with your businesses risk tolerances.



National Cyber Security Centre (NCSC)
Version 1.0
Published 25 March 2025

# What is a PAW?

**A privileged access workstation (PAW) is a trusted physical user device that is used to protect high-risk accesses from compromise by an attacker.**

A PAW aims to minimise the attack surface of the device used for high-risk accesses, making it much more difficult to compromise. At the simplest level, PAWs shouldn't be *directly* exposed to high-risk functions that could directly affect the integrity of the PAW. But if access to these functions are required, including for example email services or web browsing, it should be in a careful and constrained way. A PAW should incorporate a range of robust defences to suit your organisation's needs and the threats you face.

A PAW is a physical user device, but it is just as important to consider the supporting infrastructure and the systems to which it connects, as they have an impact on its overall effectiveness and security.

# Key concepts/glossary

This guidance uses generalised terms, which are defined below for this context:

- **Accesses**: The ability to influence a system by possessing the right credentials, connectivity and being present in a physical or logical location.
- **General access**: Where a user has no privileged roles active on their account – the expected default for the majority of users in an organisation are likely to use.
- **Privileged access**: Where a user has the ability to perform actions that a general non-privileged user cannot, such as configuring services, modifying settings or installing programs.
- **High-risk access**: A subset of privileged accesses which could have a critical impact on your organisation if misused or compromised. This includes actions that can change, remove or bypass security and safety controls.
- **Break-glass:** Highly privileged accounts and processes which bypass your normal controls and which are only used in an emergency. In this guidance, break-glass is used in the context of recovering the PAW system, not the upstream systems from which you may use a PAW to connect.
- **Corporate apps**: Applications used by the business to function, for example email and document collaboration services.
- **Threat context:**
- **Obsolete products:** Software and devices which are out of date and no longer supported and typically no longer receive security updates.
- **Managed service provider (MSP):** Any entity that delivers services, such as network, application, infrastructure and security, via ongoing and regular management, support and active administration. This may be on a customer's premises, in the MSP's data centre (hosting), or in a third-party data centre.

# The principles

## 1. Establish your organisation's PAW strategy

To design your PAW to be an effective security control, you first need to understand how it will fit into your organisation's existing privileged access management (PAM) strategy. Each organisation is unique, with a different set of threats, risk tolerances and access requirements.

You should consider which use cases and accesses your PAW requires, and how to design it in a way that is appropriate for your wider threat context. A good understanding of the threats to your business, as well as your risk appetite, helps you establish which types of accesses are high risk and should be secured with a PAW.

This principle helps you understand how a PAW complements other privileged access management approaches, and where it provides unique benefits.

## 2. Design your PAW solution to be usable and secure

An effective PAW solution is designed to meet both your organisation's user needs and its risk tolerances. Although a PAW can be highly restrictive by nature, it still needs to be an **enabling** technology. It should provide users with the tools they need to carry out their work effectively, or they will seek less secure alternatives. To ensure you have a secure solution that is also useable in practice, you should take the time to understand these needs. Designing your PAW to balance user needs and risk makes it less likely that people look for less secure workarounds.

As the needs of your organisation and users change over time, the design and implementation of your PAW should evolve too. You will need to revisit and

update both for the PAW to stay effective and to help prevent use of insecure workarounds.

## 3. Establish a foundation of trust

A PAW has some of the highest levels of access and permissions in your organisation, so it's important to build a strong foundation of trust in it, both at the start and throughout its lifecycle. Not doing this could later undermine its security controls.

As part of this, it's also crucial to consider the supply chain for all of the components, software and services used in your PAW solution. You should think about how you gain effective control and oversight when you design the PAW solution, and how to maintain it throughout its lifecycle.

You should use a 'clean' environment to build trust in your PAW solution. A building-block approach is best here, starting with a clean standalone initial device, before then rolling out and operating the PAWs across your organisation at scale.

Starting small helps you establish a foundation of trust on which to build.

## 4. Scale the solution

Where a PAW solution is made up of multiple devices, it's important to be able to scale your security controls effectively. This requires a well-secured management platform that is administered from a system you trust.

You should make sure that any modifications or changes to your PAW are consistent and reliable, for example, by using Infrastructure as Code (IaC). This allows you to automate the provisioning and configuration of your infrastructure, which reduces human error. This approach also makes it easier to duplicate environments, streamline updates and maintain compliance across your estate.

You should maintain a single view of device compliance across your estate, and closely monitor any configuration changes to make sure they are authorised.

## 5. Reduce the attack surface

A PAW device should be configured to meet your organisation's administrative access needs, while also minimising its attack surface. The goal is to mitigate risk as much as possible on your PAW, while ensuring that administrative tasks can still be carried out effectively.

You should carefully consider every feature, application and connection to a PAW and make sure they are adequately protected. Disabling unnecessary functionalities or connections helps prevent a threat actor exploiting them.

Any component of a system that connects externally can pose a threat. For this reason, you should only allow external connections when access is essential, and manage it very carefully.

It shouldn't be possible to **directly** access any services on the PAW that pose a risk to it. This includes corporate applications, email and communication tools. If you require access to these services, you should make sure you manage it carefully, so that it doesn't affect the integrity of the PAW. Examples here may include use of isolation and cross domain solution (CDS) technologies.

It's important that these controls apply to **any** device used to access high privileged or critical services, including when that use is by a third party, and you should make sure that suitable controls are in place for these users too.

## 6. Isolate high risk activity from your PAW

Maintaining trust in your PAW is essential but sometimes it may be necessary to enable software or features on it that could undermine its security posture and create additional attack surface. Examples here may include running

legacy software that is potentially vulnerable, or allowing local system administration to configure the device.

You should avoid this as far as possible, but if it's really necessary, you must fully understand the risks of doing so. An alternative is to implement these actions in a way that isolates it from the PAW. You can use virtualisation to achieve this.

## 7. Put in place protective monitoring

If your PAW is attacked, compromised or abused, it's important to be able to detect and respond accordingly. Implementing protective monitoring and auditing is a crucial part of maintaining trust in your PAW and the wider systems accessed from it.

If you have in place a good PAW strategy and effective system administration, you should have a good understanding of what actions are allowed, or not allowed, on a PAW device. This understanding will make it easier for you to put in place misuse case detections.

## 8. Control data entering and leaving the PAW solution

In some situations, you may need to import files to your PAW from an untrusted third-party location. If this data is malicious, it affects the integrity of your PAW.

Similarly, you may need to export data from your PAW which risks sensitive data leaving your device and getting into the hands of an adversary.

This presents a data import and export challenge to your organisation. Failure to put in place effective data transfer measures can drive the adoption of shadow IT or other poor practice.

# 1. Establish your organisation's PAW strategy

To design your PAW to be an effective security control, you first need to understand how it will fit into your organisation's existing privileged access management (PAM) strategy. Each organisation is unique, with a different set of threats, risk tolerances and access requirements.

You should consider which use cases and accesses your PAW requires, and how to design it in a way that is appropriate for your wider threat context. A good understanding of the threats to your organisation, as well as your risk appetite, helps you establish which types of accesses are high risk and should be secured with a PAW.

This principle helps you understand how a PAW complements other privileged access management approaches, and where they provide unique benefits.

## 1.1 Understand your organisation's strategy for privileged access management

Before implementing a PAW, you first need to understand when and why you need it. Putting in place a PAW should be part of a risk-based approach which considers what could go wrong and which attacks you need to defend against.

For some privileged access use cases, an organisation may need to make choices about how to appropriately protect them. Privileged access management (PAM) is the branch of cyber security that addresses these protections and you should consider the use and implementation of PAWs as part of a wider PAM strategy.

When looking at your PAM, consider the strengths of a PAW solution, which include how it:

- minimises the attack surface – PAWs are particularly effective to protect against an external actor compromising a device
- prevents common attack vectors, such as phishing
- prevents accidental misuse of the device, such as a user of the device inadvertently installing a malicious application
- prevents lateral movement from other devices – the reason why the interactions between PAW activity and other business activity should be minimised

PAW devices can also play an important role supporting other identity and access management controls.

Use of architectural strategies, including zero trust, doesn't remove the need for a PAW. There are some risks that can only be mitigated with a highly trusted device. A PAW underpins your technical controls, by helping to maintain trust in them.

## 1.2 Identify your high-risk accesses

**PAWs can be used for all privileged accesses but it is most important to use them for high-risk accesses.** An access is considered high risk if the potential impact of its compromise is serious, or if the systems which it protects could be of interest to a capable threat actor.

High-risk accesses are a subset of privileged access. A system or device is considered high risk if existing security controls can't sufficiently mitigate its misuse. This includes accesses that can directly modify or bypass critical security controls, or expose sensitive data, and where the consequences of which could have a serious impact on your organisation. For example, an attack that targets a vital trusted component, such as a certificate server, could undermine the integrity of your systems and be very difficult to detect and recover from.

To identify which accesses are privileged or high risk, consider what a threat actor could do if they compromised that access, including what they could do with any credentials they find, or if there is connectivity to other systems, either physically or via a network. You should pay particular attention to whether it could allow an attacker to access internal systems with weak security defences, as they could be easy targets for onward attacks. The NCSC has guidance on threat modelling which can help here.

Organisations that face a heightened threat, such as those in critical national infrastructure (CNI) sectors, are likely to have systems and devices that are of interest to highly skilled threat actors. These actors may have the skills to launch further exploits after an initial compromise as part of a multi-stage attack. If this is the case for your organisation, you should apply a PAW to a wider range of privileged accesses as part of a layered defence model. Adding defence-in-depth raises the 'cost' for a threat actor and makes launching an attack more complex.

A PAW assumes that an authenticated user is trusted, so it doesn't fundamentally mitigate insider threat. But using a PAW supports broader access management, monitoring and auditing controls which help mitigate insider risk. Logging and auditing controls is covered in more detail later in this guidance in principle 7.

# 2. Design your PAW solution to be usable and secure

An effective PAW solution is designed to meet both your organisation's user needs and its risk tolerances. Although a PAW can be highly restrictive by nature, it still needs to be an **enabling** technology. It should provide users with the tools they need to carry out their work effectively, or they will look for less secure alternatives. To ensure you have a secure solution that is also useable in practice, you should take the time to understand these needs when designing your PAW.

As the needs of your organisation and users change over time, the design and implementation of your PAW should evolve too. You will need to revisit and update both for the PAW to stay effective.

## 2.1 Design for usability and accessibility

A PAW solution changes ways of working for your users. You should design it with this in mind, taking into account the views of those who use PAWs in their work and not just the views of your risk owners. When you set up a new process, collaborate with your users, and don't make assumptions about how roles are carried out.

To minimise issues at adoption, carry out user testing and research throughout the process, especially during the early design phases. You should also routinely review the design as the requirements change, as detailed in principle 2.3 'implement lifecycle management'.

### Factors to consider during commissioning and procuring

Working to the wrong requirements at this stage risks buying or building the wrong thing. Things to consider here include:

- what works well now, and what doesn't
- who will be doing what, and under what conditions
- understanding work as it is done or prescribed, rather than work as imagined
- using diverse sources of information to gain a better understanding
- scoping usability work, so you have a better chance of getting the usability resources you need
- having suitably qualified and experienced people to carry out usability work, where possible
- not relying only on what people *say* they want, to avoid designing a 'faster horse'
- not relying only on technical excellence as this may lead to a 'perfect' control that is never used
- identifying the key people and stakeholders, including system owners, administrators, installers and end users

A solution which doesn't meet user needs will result in workarounds or shadow IT that expose your organisation to more risk. Establish a learning and feedback cycle so that you learn where processes haven't worked effectively and to help improve the system.

## 2.2 Understand your resilience requirements

Once in place, a PAW should be the **only** way of using high-risk accesses to manage systems in your organisation. You should design your PAW solution to be resilient enough to support your systems, in both normal operations and failure scenarios. In the design phase of a PAW solution, consider how resilient it needs to be and the impact if all, or part of it, fails.

Make sure that your PAW is accessible and available to the people who need it, when they need it. As you think about how to make your PAW resilient, also consider its role in the response and recovery of the systems it manages. A PAW solution is an important part of your business continuity and recovery plans. This includes acting as a trusted device when you need to establish

emergency access using 'break-glass' accounts. By making sure the solution is highly available, you reduce the need for users to resort to non-PAW devices to access emergency break-glass accounts.

There are typically no technical restrictions on the use of break- glass or emergency-access accounts, to ensure they always work when you need them. But, where possible, break-glass accounts should be used from a trusted PAW device, because of the high-risk nature of these accesses.

In exceptional circumstances where a PAW is not available and it is business-critical that access or a service is restored, your business policies should allow any device to be used to access the break-glass account.

You should also think about how you would recover the PAW solution if there is a cyber incident or system failure. This should include emergency-access accounts and procedures, availability of devices for incident investigations, as well as backup solutions.

Some key considerations for break-glass accesses are:

- A break-glass access should be exempt from the technical enforcements outlined in this guidance. It shouldn't rely on other services to be available when required.
- Within a PAW solution, the break-glass access should only be able to access the systems required to recover functionality. It shouldn't be possible to connect to the systems which the PAW administers.
- As soon as a break-glass access is activated, it should automatically trigger an alert for immediate investigation and be handled as a maximum criticality incident. Break-glass access should **only** be used as a last resort and never for any other purpose other than recovery.
- Activating the break-glass access should **always** result in post-access investigation. After the incident, make a risk-based decision to determine if trust in the solution is broken, and if any of the systems must be rebuilt, to reestablish trust and restore to a known good state. As part of the post-access investigation, you should consider how to

reduce your reliance on break-glass accounts: for example, whether you should change any other systems so that the same incident in future wouldn't require use of a break-glass account. You might also need to review the security of your break-glass access, including changing any passwords that were used during recovery.

**Break-glass access should only be used in an emergency**. Organisations or third parties should not use a break-glass access as an alternative way to gain remote access to a solution. Using a break-glass access when there is no emergency exposes excess privileges unnecessarily and makes it harder to spot a major compromise.

The credentials for the break-glass access route should be stored securely, including as a physical copy. You should make sure that only authorised individuals have access to the credentials and that access is monitored and audited.

A break-glass access should be routinely tested to make sure it still works as intended. Testing should also validate the alerting and incident escalation process.

## 2.3 Implement lifecycle management

As the technologies and user requirements in your organisation evolve, the PAW solution must evolve too. To support this, put in place an effective lifecycle management process to provide assurance that the PAW solution is effective, from its initial deployment through to end of life.

A key element to achieving this is to make sure you have an established process for change that allows a PAW user to request new functionality. There should be a decision process which establishes if the change is required, and what risks it could introduce to the system. This process should be low-overhead for the user and implemented as a continual process.

A PAW device should be an enabling technology for users which is secure, but which lets them carry out their roles effectively. Managing change effectively helps mitigate the risk that users find alternative ways to carry out their role.

As well as allowing change for new functionality, you should also provide a way to feed back on usability. The feedback process should establish if the solution still meets the needs of your organisation and users. Regularly review the PAW solution to make sure the designed functionality is still required and that it's still relevant to the systems it supports. This helps minimise the attack surface and ensures that any unused functionality doesn't remain on the device.

# 3. Establish a foundation of trust

A PAW has some of the highest levels of access and permissions in your organisation, so it's important to build a strong foundation of trust in it, both at the start and throughout its lifecycle. **Not** doing this could later undermine its security controls.

As part of this, it's also crucial to consider the supply chain for all of the components, software and services used in your PAW solution. You should think about how you gain effective control and oversight when you design the PAW solution, and how to maintain it throughout its lifecycle.

You should use a 'clean' environment to build trust in your PAW solution. A building-block approach is best here, starting with a clean standalone initial device, before then rolling out and operating the PAWs across your organisation at scale.

Starting small helps you establish a foundation of trust on which to build.

## 3.1 Establish effective control and oversight of your supply chain

Establishing trust in your PAW solution starts with understanding its supply chain and the components used within it. The NCSC supply chain security guidance provides a set of principles for all stages of procurement. You should pay particular attention to:

- **End user devices**

  You should only procure devices for your PAW solution which support hardware-backed security features. This allows you to make use of the latest root of trust, confidential computing and cryptographic technologies on your PAW. The NCSC has separate guidance to help you both securely choose devices and purchase them.

Buy your devices from trusted suppliers, and use manufacturers with good supply chain security to make sure you receive authentic devices. Part of this is considering where in the supply chain the device is assigned and attributed to your organisation, as some suppliers may pass end-customer details to the manufacturer (especially if you are planning to use services such as zero-touch enrolment.) These steps make it more difficult for an attacker to target your organisation through a supply chain attack.

- **Software including the operating system (OS) and firmware**

All software, including the OS and firmware, must originate from a trusted source. You should only use official sites or repositories to download software. To import data into your PAW network, using a file transfer mechanism helps avoids direct web access. Where possible, your import process should scan for viruses and validate the package cryptographically, to ensure that the data has not been modified before it reaches you.

- **Managed services and maintenance**

If you are using a managed service provider (MSP) to manage your PAW solution on an ongoing basis, the MSP will have **full access to all of your systems.** You should only consider doing this if you can trust the MSP and are confident they have in place strong cyber defences.

The NCSC has supply chain guidance that helps you manage MSP access.

If it isn't feasible to manage your own PAW solution in-house, it's **critical** that the MSP only manages your PAW from a device you trust, such as your own PAW, and that separation between your organisation and their other customers is maintained. This helps reduce the impact spreading, or 'blast radius', if another of their customers is compromised.

> Any dependency between your PAW and the MSP infrastructure creates a new attack vector, which could allow an attacker to pivot into your network if the third party is compromised. Where possible, these dependencies should be removed to mitigate this risk.

Identity services should **never** be shared and all MSP users should use accounts administered by your organisation. These accounts should be managed using normal business process, including a joiners, movers and leavers (JML) process. This gives you effective oversight of the users who have management privileges of your PAW solution. This is important because if you don't have assurance here, your technical controls could be undermined and you can't establish trust.

The MSP should only be able to carry out the required actions to administer the PAW solution itself. Your organisation should monitor and audit all connectivity from a MSP to ensure actions are both legitimate and planned. An MSP must access the solution in a way that allows actions to be attributable to a user, and they must not use any shared access.

## 3.2 Implement a standalone device to build your solution

It's essential to secure a PAW solution in the initial set-up phase. To minimise the risk of cross-contamination and maintain integrity, avoid building a PAW system using existing systems. Instead, begin with new and clean devices procured through a well-understood supply chain.

In small and single deployment scenarios, the initial device might serve as the sole PAW device. To make sure it functions effectively as a PAW, it should adhere to all necessary lockdown policy requirements. Typically, this initial device is used to provision your PAW management environment, creating a clean and physically separate segment of your network. This segment can then be expanded to include additional PAW devices and supporting services.

If your solution requires more than a single PAW device, the initial device should eventually be integrated into your PAW mobile device management (MDM) software, and enrolled as a PAW device. You can find out more about this in principle 4.

It's important not to use this standalone device for administration until all technical controls and policy lockdowns are in place. Once an MDM is configured, this initial device should be enrolled so that the policy is applied to all devices.

# 4. Scale the solution

Where a PAW solution is made up of multiple devices, it's important to be able to scale your security controls effectively. This requires a well-secured management platform that is administered from a system you trust.

You should make sure that any modifications or changes to your PAW system are consistent and reliable, for example, by using Infrastructure as Code (IaC). This allows you to automate the provisioning and configuration of your infrastructure, which reduces human error. This approach also makes it easier to duplicate environments, streamline updates and maintain compliance across your estate.

You should maintain a single view of device compliance across your estate, and closely monitor any configuration changes to make sure they are authorised.

## 4.1 Enforce administration from high trust

A PAW solution relies on a set of policies to determine the technical controls implemented on each PAW device. These controls are critical to the overall security of the solution, as they determine what the device can do, the features it has and the systems it can talk to.

To trust these controls, you must be able to ensure their integrity and maintain an effective audit trail of any changes made. It's also important to limit **who** can alter them. Your organisation should have a defined list of PAW administrators and within that, the number of individuals who can alter the PAW solution should be highly constrained and regularly audited. No user outside of this group should be able to make changes.

While 'browse up' refers to an architectural anti-pattern in which a system is administered from low to high trust, it is essential that a PAW follows the concept of '**browse down**', where a high-trust system administers a system of equal or lower trust. This means the PAW controls must **only** be managed from another PAW device. But consider carefully how you will avoid locking administrators out of the solution.

To avoid locking administrators out of the solution, it's possible to use a technique such as deployment pools. Here you apply policies to a set group of devices, and have a defined delay between each group receiving the updated policies. The controls and policies being applied should be tested before you deploy them at scale, to prevent major outages.

## 4.2 Implement auditable and validated controls

A set of auditable and validated controls allows you to gain assurance that each device in your solution is compliant, so you can establish confidence throughout its lifecycle, and not just at initial deployment. The controls for a PAW solution should be centralised to ensure that:

- **it is possible to enforce consistent control of policies across a deployment**
- **there is a 'single source of truth', namely a definitive overview of the policies applied to each device**

When scaling a PAW solution beyond a single device, consider using a mobile device management (MDM) service. The more devices you have, the easier it

is to use an MDM to manage them, as it provides the tools to remotely control, monitor and enforce policies on your PAW devices.

Some organisations require multiple segregated identities, such as an identity for operational technology (OT) and another for information technology (IT). A PAW solution should be designed so that it avoids introducing connectivity between these identities, as this could allow a threat actor move laterally between them.

To minimise the risk of this happening, use a separate MDM to manage the PAWs in both the OT and IT environments.

By its nature, the MDM solution has considerable privilege. It can alter the configuration, install applications and execute scripts on your PAW devices, so it's important to choose and implement a secure one. See the NCSC guidance on MDM for more information.

If the threat model of the system you are administering requires the system to be fully isolated from public and external networks, a cloud-based MDM may not be suitable, and an on-premises solution is the preferred option.

An on-premises solution should still support audited and validated controls on the device and not drive standalone or individually built devices.

When choosing an MDM for a PAW solution, consider whether it supports device attestation. This provides stronger assertions of device compliance and device health, by reporting a set of trusted signals and measurements for the device and the software running on it.

Stronger forms of attestation typically combine hardware-backed key stores and roots of trust with public-key-based cryptographic operations. The use of hardware-backed root of trust allows the measurements to be trusted, even if the device is compromised. The higher level of confidence that device attestation provides is critical for high-trust solutions, such as a PAW device.

## 4.3 Change management

Where possible, use automation to replace manual administrative processes. This could include using Infrastructure as Code (IaC) and a secure continuous integration and continuous delivery or deployment (CI/CD) mechanism. Where automation is used for configuration management and infrastructure deployments, it becomes easier to detect malicious activity using manual processes.

Use of IaC alongside a secure CI/CD mechanism can enhance your ability to scale and make changes to deployments rapidly. This also allows for effective audit trails over the PAW infrastructure, giving you greater confidence in the system and any changes made. The ability to repeatably deploy infrastructure also makes your PAW solution more resilient.

Effective permissions should limit which users, systems and solutions can alter PAW controls. Monitor for changes so that any controls in your core PAW solution are authorised, and reflected in your updated design and requirements. All changes to the PAW solution should go through the appropriate change management processes in your organisation, and require approval before being actioned.

# 5. Reduce the attack surface

A PAW device should be configured to meet your administrative access needs, while also minimising its attack surface. The goal is to mitigate risk as much as possible on your PAW, while ensuring that administrative tasks can still be carried out effectively.

You should carefully consider every feature, application and connection to a PAW, and make sure they are adequately protected. Disabling unnecessary functionalities or connections prevents a threat actor exploiting them. Any component of a system that connects externally can present a threat. For this

reason, you should only allow external connections when access is essential, and manage it very carefully.

It shouldn't be possible to directly access any services on the PAW that pose a risk to it. This includes corporate applications, email and communication tools. If you require access to these services, manage it carefully, so that it doesn't affect the integrity of the PAW. Examples here may include use of isolation and cross domain solution (CDS) technologies.

It's important that these controls apply to **any** device that is used to access high privileged or critical service, including when that use is by a third party, and you should make sure that suitable controls are also in place for these users.

## 5.1 Block access to corporate apps

It's essential to block **direct access** to corporate applications on a PAW device, including email and messaging systems.

**Direct access** in this context refers to when an application runs on the PAW, within the PAW operating system. When allowing **indirect access** to corporate applications, you should implement technical controls such as:

- cross domain solutions (CDS)
- remote desktop solutions
- a local hypervisor

The control you put in place should provide a layer of separation between the PAW device and the service being accessed. When choosing your control, consider the risk to the service and the wider threat context of your organisation.

While technical solutions that add a layer of isolation are there to improve security, misconfiguring these services can lead to a threat actor finding an exploit path between the PAW and untrusted applications. The level of

protection required to ensure adequate isolation between a PAW and untrusted services depends on your organisation's risk tolerance and the wider threat landscape.

Effective isolation is **crucial** for mitigating the most common ways an attacker gains access to systems, such as phishing or other social engineering attacks. Creating a strong boundary prevents malicious content reaching the PAW, both directly from outside the organisation, and indirectly via other corporate systems and repositories.

## 5.2 Minimise connectivity to external networks and internet services

In most use cases, a PAW requires connectivity to external networks or internet services. But its access to external services must be restricted to the services that are essential for it to operate. This may include receiving updates or configurations from an MDM, or carrying out administrative tasks on a cloud portal.

By removing access to unnecessary internet services and enforcing strict network traffic rules, you limit a threat actor's initial access and make data exfiltration more difficult.

A PAW should have technical controls in place to prevent unnecessary network connectivity. The controls implemented should depend on the accesses to which it is connecting, as well as your organisation's threat context and resiliency requirements. It's common for controls to be implemented in one of two ways:

- **Locally on the device,** such as through a software firewall with policy controls. These controls are typically static, providing a set of permanently enabled connection routes to the required services. When using local controls, users must not be able to change the rules and

policies to enable broad internet access. You should use automated auditing to make sure that internet access remains disabled.

- **Off-device or remote controls,** such as through secure access services edge (SASE). These controls typically use local software to securely connect to the SASE service, where a policy function can then be applied to the traffic. These types of services can be useful when connecting multiple sites together, and can provide a single internet edge for all devices, making lockdown policy and audit functionally easier. They also enable a more dynamic approach to controls, as connectivity routes are only enabled when required,, rather than permanently enabled. As with any third-party service, you should also consider the supply chain risks associated with any SASE products you use.

The control you choose should be well designed so that it works as intended. Think about how a threat actor could bypass these controls – for example, by using the device proxy settings to restrict internet access. Although this would restrict a device's ability to connect to the wider internet, it could easily be defeated by a DNS poisoning attack.

If your organisation is in a regulated sector, you must also consider any legal and regulatory requirements where failure or loss of connectivity to the SASE service would disrupt your ability to manage your network(s).

You should consider if the SASE product offers appropriate resilience for your regulatory requirements.

Where appropriate, you should still put in place contingency plans. This includes having a defined backup route to your services, so that privileged management can still be carried out from trusted PAW devices.

This backup route should be treated as a break-glass solution, and controlled and monitored appropriately.

## Connectivity technology

To reduce your attack surface, make sure the connectivity methods the PAW uses, such as wifi or cellular, are adequately secured.

Public wifi networks can present a major risk to a PAW device, as they often require use of a captive portal, where the local device connects directly to the local service to authenticate. The requirement for an unsecured connection to the captive portal conflicts with the on-device controls and would require an exception to a critical PAW control. **For this reason, you shouldn't allow connectivity to a captive portal from a PAW.**

If 'on-the-go' connectivity is unavoidable, you should use a corporate cellular device to provide a wifi hotspot for the PAW device. The cellular device could be a corporate phone or a mobile wifi device that allows you to authenticate with the captive portal and then acts as a bridge, providing network connectivity while ensuring there is no direct connectivity between the two devices. The PAW controls ensure that all traffic leaves the device encrypted, and there are no open services available that could be exploited from the corporate device to attack the PAW.

For organisations that use cellular networks with a private Access Point Name (APN), be aware that although an APN provides separation, it doesn't provide encryption within the telecoms network.

## 5.3 Securely connect to your administration targets

A PAW needs to communicate with the systems and devices it manages and for many use cases, this could mean multiple systems or devices. Some may be less trusted or more vulnerable than others, which presents a threat to your PAW and any other system it is used to access.

Depending on how you connect your PAW device to its target system or asset, it may be exposed to different risks. The important thing is to design it in such a way to reduce its risk of compromise, if connected to a compromised asset.

### Remote connectivity

Many use cases for PAWs involve connecting to a system over a network, either locally or remotely. It's important that communications from the PAW are protected from any threats on that network.

- You should use secure protocols, such as TLS or IPSec, to authenticate the systems or networks to which you are connecting. This also protects the confidentiality, integrity and authenticity of the data in transit. It's important that your product uses an encryption protocol that offers an appropriate strength. The NCSC has guidance on recommended profiles in its guidance on how to use TLS to protect data.
- The PAW device itself should not run any network services that accept incoming connections, and all communication should be established outbound from the PAW.
- Where you need to use insecure protocols, tunnel them over a secure protocol, or only use them on a suitably isolated network.
- For services that are considered high risk, add technical controls which ensure they can only be accessed if authenticating from the PAW device.

### Direct connectivity

Some use cases may require direct physical connectivity to the administrated device, such as over serial or Ethernet links.

When using direct physical connectivity to a system or asset, the controls on your PAW device are the only layer of security between the PAW device and the asset being managed. You should understand the risks this could introduce to your system, and mitigate them using the controls on your PAW device. For more information about controls where obsolete software is required, see principle 6: Isolate high-risk activity from your PAW.

### Isolated systems and devices

In many cases, a PAW which is used to manage standalone or isolated equipment is also used to manage or connect to other systems and networks. Where this is the case, the PAW device acts as a bridge between the systems and networks, even if the connections aren't active at the same time. Even if a network is temporary disconnected, malicious data could be stored and transferred via the PAW once it reconnects.

Where possible, avoid intentionally disabling connectivity from the PAW to its supporting systems, such as its MDM and logging solution. Removing this connectivity could limit the user's ability to carry out their role, for example, by blocking access to your file transfer solution. It also disrupts monitoring and threat detection, especially where anti-virus solutions are configured to use cloud-based analytics.

### Less trusted systems

A system or device managed by a PAW could be considered less trusted for several reasons, which include, for example:

- it processes high-risk data
- it is one of many systems which need segregating from one another
- a third party has privileged access to it

If one of these systems is compromised, you should consider the risks it would present to the PAW device and your wider estate. If you are not willing to accept this risk, you can add additional security controls between the PAW and the less trusted system. This protects the PAW device by creating stronger separation. The NCSC cross domain principles provide an appropriate level of separation. For example, you could implement a jump host (bastion host), a privileged session manager (PSM) or a hardware-based CDS solution.

A PSM can offer additional functionality for screen recording or session management, and it also keeps an audit trail of established sessions. PSMs are often provided as a part of a broader PAM product. Although they offer many

benefits, the impact of compromised privileged access is very high, so it's important to use a PAW to access the PSM solution.

## 5.4 Remove unnecessary device features, applications and functionality

A PAW aims to reduce the attack surface of the device an organisation uses for high-risk accesses. Following basic cyber hygiene measures, such as installing security updates when they are available, is an important part of this, but to reduce the PAW device attack surface further, you should remove or disable any unnecessary application or functionality.

The aim here is to design a PAW solution in such a way that it meets the requirements for your use case but ensures that each user only has access to the tools, services and features required for their role.

You should enable OS (operating system) features to restrict the execution of code only to known binaries, linked libraries or shared objects.

### Limit applications with an allow-list

Minimising access to unnecessary applications makes it more difficult to execute malicious code. Use an allow-list to limit access of approved apps to user groups or individual users. It's also important to make sure that all apps originate from a trusted source. Block any attempts to execute applications that aren't on the allow-list, and then log and investigate them.

The NCSC has guidance on using third-party applications on devices that you should apply to your PAW solution.

### Prevent remote control of a PAW device

The PAW device should represent both the physical and logical end of the management chain, meaning the point where the administrator has 'hands on keyboard'. The physical PAW device should be hardened to prevent remote

connectivity into it, by disabling associated protocols, network ports and services. Where jump hosts, privileged session managers (PSM) and virtual desktop infrastructure (VDI) are used for follow-on access, it should only be permitted from a PAW.

Sometimes it might be necessary to screen-share between a PAW and another non-PAW or external device (for example, when providing technical support). Here you should use a stronger technical solution which provides visibility of the screen but doesn't allow control of the PAW. One solution is to connect a video capture device to the PAW, which would act as a video feed a standard PC. This PC could connect with cloud-based video conferencing solutions (such as Teams), without requiring the shared device to exert any control over the PAW.

## Minimise permissions for local device users

To reduce the attack surface, the controls and features used to configure a PAW must be technically enforced. Note that having a written policy in place isn't itself a technical control. To maintain confidence in the PAW, a user shouldn't be allowed to disable the controls, as configuration changes to a PAW is considered a high-risk action. If you need to allow an engineer to amend the configuration, you need to accept the heightened risk this introduces, and ensure their actions are verified by additional monitoring and auditing.

Principle 6: Isolate high-risk activity from your PAW outlines how this could be implemented in a way that doesn't undermine the security of the PAW. Privileged accounts should not be directly accessed on the PAW.

## Only enable essential peripheral interfaces

Limiting peripheral access from a PAW is **critical** to preserve trust. Most modern devices support a wide range of peripheral connectivity, including short-range wireless connections, such as Bluetooth, and physical interfaces such as USB and Thunderbolt. These interfaces are another way that a threat

actor could gain access to a device and its data, and many of these interfaces are enabled by default.

Where you choose to enable peripheral interfaces, you should implement technical controls that limit which devices can connect. A PAW should **only** allow connections from authorised peripherals, and they should be securely sourced, and use signed drivers. For peripherals such as USB devices, policy-enforced allow-listing based on the peripherals, hardware ID and serial numbers can all be used achieve this control.

You can put in place controls that limit connectivity to these interfaces that are vendor specific. The NCSC has guidance on using peripherals securely to support the design of any technical controls implemented.

## 5.5 Protect the physical device

As well as protecting a PAW from traditional network-based attacks, it's also crucial to protect it from an attacker who has gained physical access if a device is lost, retired or stolen. Physical access makes it possible for an adversary to:

- extract cached credentials or stored sensitive data
- implant malware or modify the device configuration to undermine or bypass protections
- infect the UEFI or firmware to maintain persistence

You should add protections to the device, and put in place physical handling and storage procedures to mitigate the risk of loss, theft or physical tampering. They should be proportionate to the threat level of the systems which the PAW is used to access.

On a PAW device, it's critical to use trusted platform modules (TPMs) or similar hardware security processors to underpin device-level protections. If none are in place, many modern security features can't be used. Device-level protections must include:

- enabling secure boot so that only trusted software which is verified by a trusted authority can run during the boot process
- implementing full disk encryption to protect the confidentiality and integrity of the software and data when the device is switched off – the NCSC's device security guidance platform guides provides recommendations for your chosen platform

As well as making sure the platform is suitable to hold your organisation's privileged data, assess if data is suitable to reside on a physical end user device. You should aim to minimise data at rest on such devices because if such data persists, it heightens the risk that it may be compromised.

To minimise this risk, use a dedicated off-device storage repository for your PAW. This allows you to audit file access, user-level access controls and centralised backups of critical information more effectively.

It's essential to adequately secure the device UEFI because if a threat actor gained physical access, they could override critical device functionality and security controls. Your device UEFI should be consistently configured, locked down, monitored and securely maintained. The NCSC has additional guidance on managing device firmware which helps you choose the appropriate controls.

Finally, you should put in place a suitable mechanism to authenticate users to the PAW device to protect it from unauthorised use. Make use of security co-processors, such as TPMs, that provide hardware-backed protection of the credential on the device including brute-force credential guessing ('anti-hammer') protection. These options often also support strong MFA to administration targets using FIDO2.

Repeated failed authentication attempts should be alerted and investigated. In environments where it's necessary to enforce a strong separation of duties and have strict accountability for privileged actions, single-user devices are recommended, as it allows for stricter user-device binding.

## 5.6 Apply equivalent controls to third parties

If a third party is compromised, it offers an access route to all the organisations to which they provide services. This is why a third party access is viewed as a significant risk and should be managed appropriately.

To ensure that your PAW solution is an effective defence, any third party that requires access to your organisation's administration interfaces must also adhere to the same controls.

Where possible, supply third parties with both a PAW device and user account from your organisation. This is the most effective way to ensure you have trust in the device they are using.

But it may not be feasible in cases where a third party is providing management services to multiple organisations. Where this is the case, make sure that the PAW device which they use is built to the same security standards that your organisation has in place. Validate this either through technical or contractual controls. To build trust in the devices a third party uses, you may wish to mandate that a penetration test or risk-based review is carried out.

If you allow a third party to use their own devices, it's essential that they still authenticate through **your organisation's** identity provider. By retaining control in your environment over the identities a third party uses, you can more effectively control the policies used to validate a user's identity. This could include, for example, the locations from which a user is allowed to authenticate, or the type of MFA used.

When allowing a third party into your environment, you should carry out a risk assessment of their access, and regularly review it. Regular auditing and continuous monitoring of third party access is critical to manage the risks around it.

Only allow a third party to access equipment where there is a requirement for them to do so, and use technical controls to limit any further access.

The NCSC defines a PAW as a physical device. Any third party PAW device which connects to your administration interfaces should also be a physical device.

Allowing a third party that is working on your behalf to use virtualisation instead of a PAW is **not** a secure alternative. They must put in place a physical PAW device which complies with your organisation's polices on PAWs.

# 6. Isolate high-risk activity from your PAW

Maintaining trust in your PAW is essential but it might sometimes be necessary to enable software or features that could undermine its security posture, and create additional attack surface. Examples here include if you are running vulnerable legacy software, or allowing local system administration to configure the device.

As far as possible, you should avoid enabling such software or features but if it is really necessary, you must fully understand the risks of doing so. An alternative is to implement these actions in a way that isolates it from the PAW. You can use virtualisation to achieve this.

## 6.1 Secure your virtualisation environment

Virtualisation is not an infallible security control and attackers may target vulnerabilities in the environment to gain access to the host system. Virtualisation platforms on a PAW should minimise connectivity between the guest and host OS. As the hypervisor provides security functionality, choose a product that is robust and trusted.

To limit lateral movement, virtual environments should not have direct access to either the hosts or its networks. There are three key things to limit here:

- **Network connectivity between the virtual and host environment**

Design the virtual network in such a way that the virtual environment has no direct access to the host or other local machines. Technologies such as network address translation (NAT) can enable this, as it allows the virtual environment to access external networks by translating the virtual private IP address to the host's IP address.

Using a USB Ethernet adaptor on the PAW, and passing it directly through to the guest virtual machine, makes it possible to enable connectivity, without having to reconfigure the underlying PAW network configuration.

- **Limiting virtualisation features**

Virtualisation technology often comes with tools and features that provide integration between the PAW host and the guest virtual machine. This could include features such as clipboard sharing, file sharing and direct access to hardware devices.

Although these features make guest virtual machines more usable, they add to the attack surface, which could help an attacker 'escape out' of the virtual machine. When you enable or configure these features, be aware of these implications, and make a risk-informed decision about whether they are right for your use case.

- **Securing file transfers between the virtual and host environment**

In line with one of the core messages in this guidance, you should be thinking about useability when designing your security controls. In some use cases, there is a requirement to transfer files to the guest OS, for example, when transferring configuration files or error logs.

Where possible, data transfers should use underlying pre-built transfer tools provided by the virtualisation platform. Services to transfer data between the host and guest OS should only be enabled for the duration they are required. This lowers the risk of automated or accidental

transfers. See principle 8: control data entering and leaving the PAW solution for more about the risks to consider when designing your file transfer solution.

## 6.2 Use virtualisation to remove privileged local accounts from the PAW

Review the use of privileged local accounts on your PAW on a case-by-case basis, rather than allowing access to privileged accounts by default. Where possible, design away from patterns that require your local users to hold privileged roles on the local PAW device.

When privileged local roles are assigned, they should follow the principles of least privilege. It's possible to use virtualisation to limit this access to the virtual environment and prevent the user being able to alter the PAW device controls. Some examples of common use cases are:

- **An infrastructure administrator requiring the use of local administrator on the PAW device to execute privileged PowerShell management scripts:** In this use case, consider if a script must run on the device itself, or if it could run within a VDI, a virtual machine (following principle 5.6), CI/CD pipeline or remote code environment. This removes the execution risk from the end user device, and with it, the need for local device users to hold elevated permissions.
- **An OT engineer may require local network access to a field device for configuration, with the ability to change the Ethernet IP address of the PAW device**: In this use case, you could use a dedicated virtual machine which is built to the same control set as the physical PAW. This allows administrative permissions to the virtual environment, but not on the physical PAW device. See principle 5.6: apply equivalent controls to third parties, for more about virtualisation.
- Where legacy or overly permissive applications are driving the requirement for privileged local accounts, you should work with the vendor to identify why the application requires these permissions.

Where possible, applications which don't require privileged local accounts are preferred.

## 6.3 Virtualise obsolete applications and operating systems

In some organisations, the software required for high-risk administration relies on obsolete applications, or applications that run on an obsolete OS.

The ideal practice is that no obsolete software at all should run on a PAW. But where this is simply not feasible for effective operations to continue, primarily in OT environments, you must isolate and separate these high-risk software packages.

As a first step, engage with the vendor of the obsolete app or OS to identify alternative solutions. This could include upgrading to a newer version of the product, requesting that the application is ported to a modern OS, or seeking advice on containerisation or virtualisation of a product.

If obsolete products are still required and no alternatives routes are identified, use virtualisation to segregate them from your PAW. Also note that obsolete software should only be supplied to the users who require it. Document the obsolete products and users as technical debt, and put in place a roadmap to move to modern solutions.

The NCSC has additional guidance on managing the risks from obsolete products and joint guidance with CISA for owners and operators of OT which also covers this.

Virtualising obsolete systems can help to segregate them from the rest of your infrastructure, but a threat actor could use a compromised virtual instance to launch an attack against other virtual instances, or the underlying virtualisation platform itself.

You must treat the obsolete OS as insecure by default, meaning you can't rely on software or management of the obsolete OS to provide security functionality. An obsolete OS must have additional external controls in place.

Where the legacy virtual environment requires connectivity to external services, put in place an additional virtualised security boundary, which should be pre-configured centrally. The PAW user should not be able to make changes to the security boundaries configuration.

Within your virtualisation platform each obsolete OS should have:

- no direct network connectivity to the host machine
- no connectivity to any other virtualised hosts – each host should have its own virtual network

The virtualised managed firewall appliance should be pre-configured, and the PAW user should have no ability to change the firewall's configuration, or to log into the appliance. Any VPN configuration for the PAW running an obsolete OS should be at the firewall appliance in the first instance (and not on the obsolete OS).

It's common to see security tooling, such as third-party antivirus solutions, deployed to these obsolete VMs. You should consider any increased external connectivity risk from the virtual environment from these additional tools **before** they are implemented. As they probably operate from a vulnerable OS, their results can't be implicitly trusted, meaning an antivirus product on a virtualised obsolete OS has limited effectiveness. Instead, it's better to carry out additional external monitoring.

# 7. Put in place protective monitoring

PAW environments are inherently constrained by design, with a limited number of use cases. As you design how to monitor these solutions, make use

of the restrictive nature so that it is easier to identify anomalous behaviours which may indicate a compromise.

The NCSC has separate guidance principles on logging and protective monitoring as well as building a security operations centre (SOC) and you should make sure you apply these principles in a PAW environment.

When you are looking to put in place monitoring, your chosen solution must be able to provide visibility over both the systems that support and configure the device, as well as the device itself.

Logging events should be streamed to a log-processing system in near real-time, with protections in place to prevent a threat acter tampering with logs on the device. Your log-processing system should also be well protected to prevent tampering, deletion or stopped completely.

Logs collected from the PAW solution must be immutable to make it harder for an attacker to interfere with them or to manipulate them. There may be situations where a PAW user with administrative responsibilities requires authorised privileged access to a log collection or storage solution. But this level of access should be highly restricted and, where possible, routinely audited to make sure that logs aren't changed or amended without authorisation. Where feasible, logs for the PAW device should be sent directly to a central monitoring solution, such as your SOC.

## 7.1 Put in place monitoring for both configuration and change management systems

It's important to be able to trust the whole of the PAW solution and putting in place monitoring helps maintain confidence in its integrity.

It's particularly important to monitor both the configuration and change management systems. As any changes could significantly impact the security of the physical PAW device, for example by reenabling functionality or adding additional tooling, you should set up alerts when any changes are made. This

allows your security team to map alterations when an authorised user makes a planned change.

## 7.2 Put in place anomaly detection

Within the PAW solution, it's important to put in place additional monitoring for all uses of privileged identity, in line with the NCSC secure systems administration guidance. Where possible, you should also put in place additional logging to monitor any actions carried out on the PAW device. This allows greater accountability of actions and is particularly important to monitor the risk of insider threat in your organisation.

As a PAW has, by its nature, a constrained set of functions, you should aim to integrate anomaly detection into device monitoring. Any activity outside of these functions is an immediate indicator of a misconfiguration or compromise. Where it's feasible, correlate the logs between the PAW device and the target system to identify anomalies.

# 8. Control data entering and leaving the PAW solution

You should carefully manage data that is both imported to and exported from your PAW environment. If you can't trust files that are downloaded and then opened on your PAW device, they could be malicious and exploit your PAW.

Similarly, if a PAW is compromised, an attacker could abuse how your PAW exports data to exfiltrate sensitive information. For these reasons, you should carefully constrain and control how data is imported and exported from your PAW solution.

Building an effective process for important and exporting data is critical to enable your engineers to carry out their duties. Where this is not in place, users may resort to shadow IT.

An effective solution to manage your data effectively also helps you migrate away from the use of higher-risk transfer methods, such as removable media. Where it's not possible to remove these methods from your environment, you should still apply the controls described in this principle. RITICS has guidance on managing removable media for ICS and OT environments that can support you here.

## 8.1 Secure use of enterprise file solutions

It's important not to allow direct or unmanaged access directly from a PAW to your corporate file-sharing services. Examples of this could include mounting permissive network drives on your PAW, or directly accessing cloud-based file sharing services. Instead, make sure that importing and exporting data only takes place in line with a carefully managed import and export process.

Carrying out data transfers using your existing enterprise storage solution can present major risks to the PAW solution. These solutions are likely to introduce additional lateral movement routes between your enterprise and PAW environment. It can also be difficult to enable only the storage element of enterprise solutions, as it requires you to permit large amounts of internet-bound connections. Only use enterprise storage when you can be sure that:

- your enterprise and PAW identities remain segregated
- you have put in place mitigations to the risk of lateral movement
- enabling network connectivity to the enterprise storage solution does not allow new connectivity to any additional services

## 8.2 Effective import and export controls

When designing an import and export solution that is suitable for a PAW, start by identifying what types of data you need to share between your enterprise and PAW environments.

For example, complex data types, such as Word or PDF files, present greater risks that are typically harder to mitigate than structured data formats, such as XML or JSON. You should aim to minimise transfers to only what is essential, and where possible, consider converting to simpler data formats.

Your data transfer solution should:

- produce an audit trail so that you can account for all data entering and leaving your PAW environment

- require user authentication
- ensure only approved and authorised content is exported
- automate the transfer process to a defined pre-configured endpoint – the system should not allow export to arbitrary end points.
- inspect and scan the data for malicious content – this is partially important for data imported to the PAW. In some situations you may also want to validate the data structure.

The NCSC has separate guidance on how to safely import and export data from trusted environments as part of the CDS guidance collection. Although these patterns are most effective when fully implemented where an organisation faces a lower threat, it's possible to apply subsets of these patterns.

For high-threat use cases, data control solutions should, where feasible, be implemented in hardware following the NCSC CDS principles. Where the threat context is lower, software-based controls and network boundary restrictions may be suitable.

© Crown copyright 2025. Photographs and infographics may include material under licence from third parties and are not available for re-use. Text content is licenced for re-use under the Open Government Licence v3.0.
(https://www.nationalarchives.gov.uk/doc/open-government-licence/version/3/)