Purpose of This White Paper

What is a PKI

Public Key Infrastructure (PKI) is a system of cryptographic technologies, standards, management processes, and controls governing the use of digital certificates. More information on What a PKI is can be found here. PKIs are made up of a number of Certification Authorities (CAs) and supporting infrastructure.

What is a Private PKI

A private PKI is a PKI that is not publicly trusted. It has its trust limited to the organization or a group of organizations sharing a common purpose or goal. Limiting trust is an important and useful aspect of the solution as it ensures that an organization remains in control of who and what trusts the PKI and it's issued certificates. This in turn allows a system, service, application, device or user to restrict trust to only those systems, services, applications, devices and users that that should be trusted.

Suggestions

Below is a list of security considerations that Cogito believe are essential to assess as part of selecting or designing a service. Some of the security considerations are part of standard Cyber Security best practice and some are specific to PKI. More information on standard Cyber Security best practices can be found at in our Knowledge Articles.

Purpose of the Private PKI

Much of the design of a private PKI, and the answer to many of the options below is and should be determined by the purpose of the PKI. In particular what systems and services will it support and what is the impact of a compromise on those systems. Common uses of a PKI (also known as Public Key Enablement or PKE) can be found at Knowledge Articles.

On Premises or as a Service

Determining this can be based in large part on the organization's overarching cloud strategy. Many of today's organizations have a cloud first strategy. The considerations below can be a determining factor in deciding if this particular part of the infrastructure will be put in that default cloud platform, or an alternative approach needs to be sought. This can lead some organizations to determine that they want a better security and/or functional outcome than the default cloud service used by the organization can provide. This drives some organizations to keep their private PKI as one of the few pieces of infrastructure that will remain on premises, or increasingly to seek a specialized third-party provider such as Cogito Group, who provide a better security and functional outcome than can be achieved by an on premises or primary cloud provider's service.

Physical Controls

There are many physical controls place on a PKI's operations. Often the service is physically segregated from other services and systems. It is also operated out of separate physical zones. Cogito for instance uses Remote Management Centers (RMCs) as the only locations outside of the Cogito Datacenters where our core PKI components can be administered from. Other physical controls can be the use of no-lone-zones, non-networked computers and the use of Safes and encrypted storage medium to name just a few physical controls.

Personnel Controls - 1 or 2 Person Control

Many high security PKIs will have two-person control on PKI components and restrict the administration of the service to what are known as no-lone-zones. No-lone-zones are dedicated areas where more than one person must be present to operate a capability. Operating with a primary operator and a secondary operator acting as a witness is a core tenant of a secure PKI, as it ensures no one person can modify the system or service without at least one other person knowing about this modification. Best of breed services such as the Cogito Service implement two person controls but also back this up with multiple policy, audit and technical controls to ensure no one person or even two can make unapproved changes.

Personnel Controls - Other

The following additional personnel controls should be part of the strategy to protect a private PKI service:

- Vetting of personnel
- Limiting the number of personnel that have access to the service
- Use of least privilege controls as described below
- Auditing of the solution by third parties and a monthly audit of material and personnel activities.

Logical Controls

Zero-trust architectures is a new spin on an old idea around an east-west strategy. East-West was the idea that just because a bad actor gets into one system (from the North or Firewall) they should not be able to move laterally through the network to compromise more of that network. Zero-trust is something that many organizations are now adopting. In a zero-trust architecture nothing gets automatic trust. Every service, application, device and user must prove itself to everything it connects to. Cogito has embraced a zero-trust architecture even at a software subcomponent level. The Jellyfish platform which is the basis of our as a service offering requires every microservice to prove it's trust to every other microservice it connects to and all traffic between the microservices is encrypted. This is extended to our containers, virtual machines, switches, firewalls, physical servers, etc.

Network segregation is a key strategy that can be employed as part of a zero-trust architecture or preceding it. Network segregation provides another check point within a network and limits the compromise of an entire network should a single entity become compromised.

Multifactor authentication (MFA) provides authentication for users through the utilization of two or more authentication factors. These authentication factors may include something the person knows (e.g. a password), something a person has (smartcard, FIDO token, etc) and something the person is (fingerprint, facial biometric or iris scan).

To ensure full coverage of MFA across all services and applications it may be necessary for a user to utilize several MFA authentication methods such as a smartcard for network login and OTP or FIDO for web application authentication.

Role based access control and least privilege controls limit a user to just the capability they need to perform their role. This prevents a user's account that has been compromised from compromising the entire network.

Transition Old Infrastructure or Start Again (Greenfields)

If there are any concerns about the security of the existing system, Cogito always recommend starting again. The great thing about PKI is that two systems can easily co-exist for a period of time with no issues. You can and do often have multiple trust points within an organization already such as public trust points installed in operating systems and browsers. It is also required as part of a standard feature of PKI, which is that trust anchors have a finite life and expire. Often the only issue with running two PKI's side by side is the management overhead of doing so. There is in fact no real benefit in keeping with an older PKI old unless you have existing hardware and even then starting again is often the base option if starting from scratch

PKI Hierarchy, Number and Purpose of CAs

Determining the PKI Hierarchy that best suits the organizations needs should be based on the purpose of the CAs. What types of certificates are they likely to issue and what is the protection required on the PKI service.

Hierarchy's come in different forms. Common infrastructures come in the following formats:

- Root and issuing CA combined
- Separate Root and Issuing CAs
- Root, Policy and Issuing CA levels

More complex models can include dual trust/cross certification and bridge models.

Offline vs Online CAs

A key control for highly secure PKI's is to limit the access to critical components like CA components. One way this is done is by having Root of trust CAs (Root CAs) offline (i.e. they are not network accessible in any way). When combined with personnel controls, offline CAs become very difficult to compromise. Issuing CAs on the other hand are usually online to increase their usefulness such as allowing methods for automating certificate requests and delivery.

Hardware Key Protection

Many PKIs store CA private keys in Hardware Security Modules (HSMs). These devices provide not just strong key protection but can also accelerate the use of these keys when performing key generation, encryption and signature functions. While some HSM manufactures allow export of keys, best practice for asymmetric CA keys is to mark the private keys as non-exportable. Another best practice is to select a manufacturer that has certifications such as FIPS 140-2 or the newer FIPS 140-3.

End Entity Usage

End entity usage, such as by a user, device, application or service should drive key usages, templates required, registration methods needed, and even the design and security posture of the PKI. Any PKI is only as strong as its weakest link, so if a high level of assurance is required in the end entity certificates, then this must be reflected in all parts of the hierarchy.

Algorithm Type and Size

There are two main asymmetric algorithm types in widespread use for PKI. These are the RSA algorithem and the Eliptic Curve Cryptography (ECC) suite of algorithms. RSA is the older of the two types, but also has the most support from applications, devices and services, especially older devices.

The algorithm used is one key factor in determining how hard it is to compromise the PKI and another is key size. The larger the key the harder it is to compromise. However just selecting the largest key can have both performance and compatibility impacts on the systems to be supported by the PKI. Common key sizes for RSA keys for instance are now 4096-bit keys for a CA and 2048 bit keys for an end entity. For ECC, which is newer and has smaller bit lengths for relatively the same level of protection, a 384-bit length key pair is often used.

Software To Use

The software used provide PKI functionality, such as the Certification Authority software is critical to the reliable and secure operation of the service. The following factors are key in determining the best software for a private PKI:

- Security - Critical to the security of the PKI is that the software used is secure. There are a number of ways that this can be achieved, but one key way is to use a certification scheme such as NIAP or Common Criteria. Cogito's own CA software is currently undergoing certification under both schemes and advice on this can be found here.
- Functionality - The system should have sufficient functionality to meet the requirements of the organisation. A base CA is capable of generating certificates, but there are many features within a CA that should be considered. Some of those can be found through one example of CA software being Microsoft's Active Directory Certificate Services (AD CS) platform. Information on where AD CS is appropriate and where it is not can be found here. Other key functional requirements that should be considered include the ability to use HSMs, the number and type of automated services available to the

platform and missing in many platforms, is the ability to perform Certificate Lifecycle Management (CLM) functions. This is the rich services such as search and notification.

- Usability - Some CA and CLM platforms can be hard to set up and hard to use. The more capable a system, generally the more difficult it becomes to use, due to all of the capability available in the product. Simplicity however should still be the goal from at least a use point of view. Often a good choice to reduce complexity and the knowledge required to set up and run a PKI service can be avoided by using an as a service provider like Cogito's SecureSME service.

Ancillary Certificate Information

There are a number of important requirements that must be covered as part of a good private PKI design. These commonly include, but are not limited to:

- Publication point locations - Publication points such as Certificate Revocation List (CRL) Distribution Points (CDPs) are a key factor that must be built in at the outset, as like many other components on this list, they cannot be changed once set in a CA without recreating the trust points. Common CDP protocols include the Hyper Text Transfer Protocol (HTTP), Lightwieght Directory Access Protocol (LDAP) and less commonly File Transfer Protocol (FTP). Another key publication point is the Authoritive Information Access (AIA) addresses. This tells an application or device where to go to find parts of the trust chain, but can also be used to provide a location of confirming the validity of a certificate through the use of the Online Certificate Status Protocol. This protocol can be more accurate and less costly in terms of the speed of a transaction then the use of CRLs.
- Maximum certificate lifespan - this is the maximum life that a certificate can be issued for. This can be set on CA or end entity certificates.
- Maximum CRL lifespan - This is the maximum lifespan of a CRL. This ensures that computers relying on the certificates will come back to recheck periodically if a certificate that is being relied upon can still be trusted and used. The lower the lifespan the better the security of the system, but the more often services must download the CRL which can slow transactions.
- Maximum certificate lifespan
- Basic constraints - Basic constraints allow for limitations to be built into a PKI hierarchy. An example of a basic constraint is the ability to restrict how many CA tiers can be in a PKI hierarchy.
- Name constraints - This allows for limits to be placed on what names will be allowed to be trusted within the PKI. This must be supported by applications however, not all of which do support them.
- Support or otherwise for certificate modification - Certificate modification is where a certificate is resigned using the existing key. This is rarely supported in certificates that require the best security.
- Support or otherwise for certificate suspension - Certificate suspension allows for a certificate to come back from being revoked. It is again often not supported in secure private PKIs.

Operations Checks

Operational checks of the PKI are critical to the smooth operation of that PKI. This can take the form of automated monitoring in some cases, but can also be the implementation of daily, weekly, monthly and annual systems operability tests or checks as well. Cogito recommends the implementation of both system operability tests as well as automated monitoring with external alerting.

Glossary of PKI Terms and Acronyms

A glossary of PKI Terms and Acronyms can be found in Cogito's Knowledge Articles.