5 Steps to Building a Scalable PKI: The Security Engineer's Guide

Darin Andrew

digicert®

Table of Contents

Introduction

Tom is responsible for security engineering at his company. Sales have been growing fast, and executive leadership announced plans to invest in new projects and additional headcount. Leadership is counting on Tom and his team to enhance network security to prepare for this growth.

Now, Tom's feeling the pressure to present a plan in next week's meeting with his boss. He knows this plan will involve Public Key Infrastructure (PKI), but he wants to validate his thoughts, and find out what top enterprises have done under similar circumstances. If he can't have a plan fully developed, he would at least like to respond confidently to his boss' questions in next week's meeting.

The problem is, building a reliable PKI architecture can be complex, time consuming, and costly. In fact, only 27 percent of information security decision-makers are extremely or very confident they have the right information technologies in place, according to IDG.

So, Tom starts asking around for answers to his questions: How can we build the security measures we need to handle this growth? What new technologies will we need to adopt? Who will we need to hire to support new areas of our network security? What have leading enterprises done to mitigate similar risks?

---

27 percent of information security decision-makers are extremely or very confident they have the right information technologies in place.

IDG

If you're feeling Tom's pain, you're in the right place. In what follows, we'll help you plan and build a PKI that will scale with your business.

5 Steps to Building a Scalable PKI

Like Tom, you may stress over maintaining the trust of your executive leaders. After all, you're responsible for a critical piece of your company's business. As proof, 61 percent of CEOs worry about security's impact on company growth.

---

61 percent of CEOs worry about security's impact on company growth.

19th Annual Global CEO Survey / January 2016

So that you can ease your CEO's worries about security's impact on growth—and give him or her confidence in you as a leader—we've simplified building a PKI into five actions:

1. Identify your non-negotiable network security risks
2. Pinpoint the network security risks PKI can mitigate
3. Develop the right mix of public and private PKI
4. Decide whether to build (internal CA) or buy (hosted CA)
5. Determine how to automate delivery of certificates to devices

digicert®

1. Identify Your Non-Negotiable Network Security Risks

We'll dig into the more technical aspects of PKI starting in step three. But first, you'll need a high-level understanding of the risks you must mitigate in your business. Examples of these security risks include:

- Preventing unauthorized access to web services
- Preventing unauthorized access to knowledge stored in databases
- Preventing unauthorized access to your network
- Verifying authenticity of messages transferred on your network

Defining these basic risks first will help identify which can be solved using PKI.

2. Pinpoint the Network Security Risks PKI Can Mitigate

With PKI, you can significantly increase the security level of your network. PKI binds an identity to a public key. This allows you to mitigate risks through encryption, digital signatures, and authentication. Encryption will help you mitigate risks to confidentiality. Digital signatures will help you mitigate risks to integrity. Authentication certificates will help you mitigate risks to access controls. This can be applied to various applications.

Common PKI use cases:

- Securing web pages
- Encrypting files
- Authenticating and encrypting email messages using S/MIME
- Authenticating logins using smart cards
- Authenticating nodes connecting to a wireless network
- Authenticating connections to your VPN
- Authenticating connections to sites and services containing corporate data using TLS mutual authentication

3. Develop the Right Mix of Private & Public PKI

Once you identify your non-negotiable network security risks, and decide which of these can be mitigated using PKI, it's time to plan your PKI architecture.

Most mature enterprises have built a hybrid architecture that includes both public and private PKI. They typically use public PKI to secure their public-facing websites and services, and private PKI to secure their internal ones. They also differ on how automated their process is for delivering certificates.

To find the right PKI mix for your organization, you first need to identify where you need private PKI and where public PKI will be more advantageous. As promised, let's dig into some of the more technical PKI considerations.

PRIVATE VS. PUBLIC PKI

With PKI, you're binding an identity to the public key through a signing process. That signature is performed by a root, or with an intermediate that chains up to the root. Only certificates issued from roots you trust are recognized as valid.

If the root that bound the identity to the public key is in your trust store, then you can rely on the identity bound to the

public key (rely on the subject of the certificate). This is all because the certificate was issued by a root you trust.

So, what's the difference between a public root and a private root? When and how should you use each?

WHEN TO USE A PUBLIC ROOT

The technology used for signing a certificate is the same whether signing with a private or public root. Instead, the difference is that a publicly trusted root is already distributed out to browsers, operating systems, phones, etc. When a user tries to visit your site, her browser (e.g., Google Chrome, Mozilla Firefox, etc.) checks to see if the root that issued the certificate is on its trusted list of roots.

Let's run with the case of a webpage. Does the browser that's connecting to your web page possess the root? The answer usually depends on whether it's being accessed from a computer your organization manages or not. If your organization controls the devices, then you can distribute a private root to the trust stores there. The browser will trust certificates issued from any root that has been distributed to its trust store—even certificates issued from a private root.

But what happens if this is a public web page that anyone in the world can hit? Unless you've distributed your private root to every single device that's used to visit the page (which isn't possible), users will receive a warning message saying the certificate isn't trusted because it was not issued by a trusted root.

Browsers are giving severe warning messages these days. Either the user won't be able to visit the page, or she'll be forced to change her settings to make the connection. That's not a good place to be in.

WHEN TO USE A PRIVATE ROOT

Browsers are giving severe warning messages these days. Either the user won't be able to visit the page, or she'll be forced to change her settings to make the connection.

A strong use case for private roots is authenticating internal services. For example, a private root is useful for authenticating connections into your virtual private network (VPN), internal Wi-Fi, Wiki pages, or other services that support multi-factor authentication.

In all these cases, you control the server instance that's checking the validity of the certificate, so a private root is ideal. Your internal operations team can specify your own private root as the issuer of certificates, and when validity is being checked, it can see that it was issued by your own trusted private root.

Issuing certificates from a private root gives you more control over the issuance process, certificate profiles, and subjects named in the certificates.

The benefits of a private root for authentication boil down to control. Only your organization has the rights to issue certificates from your own private root. This gives you more control over the issuance process, certificate profiles, and subjects named in certificates.

4. Decide Between Hosted or Internal CA—Build or Buy?

Once you've identified where you need private certificates for your internal services, decide whether you should create an internal PKI (build) or use a hosted PKI service (buy).

Both build and buy are good options. The decision comes down to the resources and personnel you're able to dedicate to PKI. A hosted service creates your root and secures it at a level commensurate with public trust anchors. An internal CA gives you full control of the issuance process, but requires you to take on the costs of software, hardware, licensing, and training. We'll go into more detail about the benefits and drawbacks of each type of CA, and look at average costs of each in detail.

The real question is whether an internally managed PKI is worth the investment in your organization's time, money, and personnel. Managing an internal PKI system has both benefits and hidden costs. However, what starts as a financially viable plan can quickly turn into an economic disaster. Hardware costs alone—machines like hardware security modules (HSMs)—can easily add a quick $50,000 or so to your total investment.

Often, engineers falsely assume a commercial CA only specializes in public PKI, and that it won't provide them with cost-effective, flexible solutions for private PKI.

COMMON MISCONCEPTIONS OF HOSTED PRIVATE CA

Network engineering teams sometimes decide against a hosted CA due to common misconceptions. Often, they falsely assume a commercial CA only specializes in public PKI, and that it won't provide them with cost-effective, flexible solutions for private PKI.

Engineers often mistakenly assume a hosted CA:

- Will charge the same price for private certs as public certs
- Won't give them the flexibility to automate certificate processes
- Will limit them to certain certificate profiles

Cost. You may have only worked with a commercial CA to purchase public SSL/TLS certificates. With this as your only reference point, you might assume private certificates have similar costs as public certificates—this isn't the case. Issuing a private certificate from a commercial CA's hosted solution is typically a fraction of the cost of issuing a public certificate with that same commercial CA.

Flexibility. Another common misconception many have is that they won't be able to accomplish the same goals with a hosted solution as they could with an internal CA. For example, you might wonder whether you can automate certificate issuance with a hosted solution. Many commercial CAs have tools, like RESTful APIs, for automating certificate management. Before you choose a commercial CA, check out its platform, tools, and integrations.

Certificate Profiles. Many think a hosted CA will limit them to certain certificate profiles. They think they'll only get certificate profiles that are approved by the CA/Browser Forum. But, because these are private certificates, most CAs can provide any certificate profile you need. They typically don't have to be SSL certificate profiles—they don't even have to be X.509.

WHEN TO BUILD (INTERNAL CA)

The first thing to consider is scale. How do you determine the scale of your PKI? The most common mistake engineers make is building an internal CA based on the PKI project at hand, only to discover it's not sufficient a few years down the road. If you're not careful, you can put valuable resources into building an internal CA, but be forced to abandon the project if you can't get it to scale.

For example, let's say your current need for an internal CA is to issue authentication certs to your laptops and phones, so they can authenticate into your wireless. This will require building a fairly affordable internal CA. However, as larger projects present themselves, growing your internal CA can begin to create financial strain.

> Don't be fooled by the scope of your current PKI project—think long term.

Six months down the road, you might realize you need to set up certs for all your internal servers. You might want certificates to be issued to all servers automatically through an API. Now, you're creating an API interface—yet another project. What started as a small project, is ballooning into a resource-intensive one.

Don't be fooled by the scope of your current PKI project—think long term. It's hard to guess what you'll need 5 to 10 years from now. A commercial CA can help guide you through this. Because the commercial CA has worked with a broad range of enterprises, it will have a view into how those PKIs have scaled over the years.

You should also consider how existing resources might alleviate some of the burden of building an internal CA. For example, if you have a Microsoft Server license, this will eliminate the cost of CA server, since this is included in the cost of your license. You could also check with your team to see if you have available resources, like segregated networks, firewalls, dedicated rack space, and engineers with applicable knowledge. If you already have some of these available, it might make more sense to build, rather than buy. Later, we'll list out the costs of an internal CA in detail. Keep some of these considerations in mind as you go through that list.

You should only invest in building an internal CA once you've considered both the financial cost of the technology and the opportunity cost of your engineer's time. You'll know it's time to build an internal CA once you've verified you have

the budget and time, and that you truly need the control and customization an internal CA provides. Keep in mind, however, that a hosted solution can often offer similar benefits for a fraction of the cost.

WHEN TO BUY (HOSTED CA)

Commercial CAs have put massive resources into hardware, software, personnel, training, certificate policy, auditing, and vulnerability testing. In many cases, you'll save time and resources by leveraging the infrastructure the commercial CA has already built, instead of putting hours and dollars into building your own. If you're a smaller team with a flat budget, it may make more sense to use a hosted private PKI solution, since this provides many of the benefits of an internal CA, but without most of the costs.

Once you've decided it's best for you to buy rather than build, you'll want to do some checking to ensure the commercial CA can give you what you need. These considerations involve reliability, deployment ease, functionality, support, and cost.

Reliability. Is the commercial CA financially stable? You're going to be investing significant time and resources into working with your CA—make sure you won't run into a situation where your PKI suddenly stops working.

Deployment Ease. Does the commercial CA offer APIs that help automate certificate deployment? Can the CA deliver certificates into your infrastructure without security gaps? Is there a delay between request and issuance? Will this affect your users' ability to perform their responsibilities? These are all deployment questions you'll want to ask before deciding on a commercial CA.

Support. Does the commercial CA offer 24/7 support? This is arguably one of the most important considerations on this list. When your network engineers run into brick walls, how easy is the CA to work with? How quickly can they help your engineers solve problems so they can get back to their core projects?

Cost. What does it cost to issue a certificate with the commercial CA? Commercial CAs typically charge per certificate and validity period. Most offer different types of certificates depending on the size and need of your security landscape. Although this can seem like an expensive option, commercial CAs have invested countless hours and resources into building the infrastructure needed to deploy both public and private certificates. As mentioned, commercial CAs typically offer private certificates at a fraction of the cost of their public certificates.

COST COMPARISON: HOSTED VS. INTERNAL CA

The cost of an internal CA varies widely depending on the scope of your projects, number of certificates, and need for software, hardware, personnel, redundancy, certificate policies, and vulnerability testing. Because of this variation, it's difficult to provide a universal number for the cost of an internal CA.

Although we can't predict each cost needed for your specific PKI, we can list out the typical resources needed. With the following list, you can verify whether you already have the resource, and, if not, how much it will likely cost if you don't.

The costs break down into six major categories:

1. Hardware, software, and licensing
2. PKI expertise
3. Training
4. Certificate policy (CP) and certificate practices statement (CPS)
5. Auditing against certificate policy
6. Vulnerability testing

COSTS OF AN INTERNAL PRIVATE PKI

Hardware, Software, and Licensing	• CA server—included with Microsoft Certificate Services (2 recommended for redundancy) • Online Certificate Status Protocol (OCSP) and Certificate Revocation List (CRL) distributed services for redundancy, high availability, and fast response times • Firewalls and segregated networks (Firewall, switch, and dedicated rack space) • Storage mechanism for offline root and backup of offline root (HSM required) • Signing HSMs—Gemalto Luna 5 ~$40k-60k (2 recommended for redundancy)
PKI Expertise	• PKI authorities and admins (2 for separation of roles) • Developer to write API interface (If customization is needed) NOTE: Average industry salary: $120-200k/individual
Training	• Regular training to keep personnel updated on latest PKI changes • Courses, certifications, and conferences
Certificate Policy (CP)/ Certificate Practices Statement (CPS)	• See most up-to-date reference (RFC 3647) for details: https://tools.ietf.org/html/rfc3647 • Writing a CP/CPS (80+ hours of work for PKI staff) • Maintaining a CP/CPS (living docs that need to be kept up-to-date) • Enforcing CP/CPS in software, policies, and rules
Auditing Against Certificate Policy	• On-going logging of key portions of PKI as evidence for audit • Yearly audit of check compliance with policies in CP/CPS
Vulnerability Testing	• PEN testing for CA and supporting services—$40-60k/pen test (Recommended on a regular basis—frequency defined in CPS) • Auditing vulnerability compliance, network scans, and vulnerability scans

BENEFITS OF A HOSTED PRIVATE PKI

Costs Avoided by Using a Reliable Hosted Private PKI

- Trained personnel to securely manage the CA
- Hardware, software, and licensing
- Industry updates in servers, browsers, and libraries
- High-availability and revocation infrastructure (OCSP & CRLs)
- Certificate management via API

One cost engineers often overlook is the cost of personnel. Not just the cost of hiring additional personnel to build and manage the internal CA, but the opportunity cost of your engineering team's time.

One cost engineers often overlook is the cost of personnel. Not just the cost of hiring additional personnel to build and manage the internal CA, but the opportunity cost of your engineering team's time. If they're putting hours in to build an internal CA, they're taking hours away from their core projects.

Your engineering team has many other responsibilities for security and maintaining infrastructure—like email servers, wireless, penetration testing, audits, risk assessments, and the list goes on.

Changing industry standards and shrinking certificate validity periods mean automation won't be an option in the future—it'll be a necessity.

5. Automate Certificate Delivery

For your PKI to run smoothly at a large scale, you'll need to automate certificate deployment. Changing industry standards and shrinking certificate validity periods mean automation won't be an option in the future—it'll be a necessity. You might oversee hundreds or thousands of devices. Leveraging automation will make your team more efficient, and help you maintain security by reducing human error and certificate-caused outages.

You have four major options for automation:

1. RESTful API
2. Simple Certificate Enrollment Protocol (SCEP)
3. Enrollment over Secure Transport (EST)
4. Microsoft AD Auto-enrollment

RESTFUL API

As we mentioned, it's important to verify whether the commercial CA you've chosen offers APIs for you to program against. Does the CA allow you to use RESTful API endpoints as part of the programming you're doing on your side?

Are you already using enterprise device management software (Tools like Venafi, AirWatch, Casper, Tanium, etc.)? If so, search for a commercial CA that will allow you to deliver certificates to your devices by integrating with these software solutions you're already using to manage devices.

SCEP

This route requires a SCEP agent on the device, and works in conjunction with enterprise device management software. The software sends the script down to the device, telling it to go get a cert and providing the configuration information to hit the SCEP service.

Then, the SCEP service takes it from there to get a cert on the device. The benefit of this is that any device that supports SCEP (Android, Microsoft Windows, Apple iOS, and other operating systems support SCEP agents) makes it quicker to go from proof-of-concept to production because it's already an established protocol.

The benefit of SCEP is that the agent already knows how to deliver certificates to the device. The agent will automatically put it into the operating system's key store. Some enterprise device management systems have this capability, but this is a question you should ask your software provider. If your software has this capability, something like the DigiCert RESTful API could take the place of the SCEP agent.

EST

As the successor to SCEP, EST is almost identical, except that it supports Elliptic Curve Cryptography (ECC). ECC is simply a type of cryptography that creates faster, smaller, and more efficient cryptographic keys.

MICROSOFT AD AUTO-ENROLLMENT

This can be used for automating certificate delivery to the Microsoft Key Store on all Windows PCs and servers. If you've already been using Microsoft AD for other purposes, it might make sense to use AD Auto-enrollment for certificate automation.

DigiCert Managed PKI

Now that you have a clear plan for building PKI into your network, DigiCert can help you make it happen with solutions for both public and private PKI.

CLOUD CA	
Private SSL	Get stronger oversight with a Dedicated Intermediate that is branded and accommodates custom profiles.
Public SSL	The DigiCert Cloud PKI service accommodates high-volume deployment for certificates trusted by all major browsers, devices, and operating systems.

INTERNAL CA	
Private SSL	Issue internally trusted certificates from an in-house private Issue internally trusted certificates from an in-house private PKI.
Public SSL	Unavailable

PRIVATE PKI

DigiCert offers both a hosted and an internal solution for private PKI. Our expert PKI architects can help you customize a solution for your specific environment. Whether you're ready to use a hosted solution, or are still evaluating this decision, our expert engineers can help you reach a decision—whether that's to build or to buy.

DigiCert Cloud CA. Our hosted solution lets you keep the control with none of the maintenance frustration. We'll create your root and secure it at a level commensurate with public trust anchors, while giving you oversight of your intermediate, its properties, the types of certificates it can issue, and the names on those certificates.

Benefits:

- Trained personnel to securely manage the CA
- Hardware, software, and licensing
- Industry updates in servers, browsers, and libraries
- Revocation infrastructure (OSCP and CRLs)
- Certificate management via REST API

DigiCert Internal CA. Issue internally trusted certificates for your organization from an in-house private PKI.

Benefits:

- Full control of issuance
- No internet dependency
- Configuration/Dev changes done on your schedule

AUTOMATE & CUSTOMIZE WITH DIGICERT RESTFUL API

As mentioned, the DigiCert RESTful API allows you to integrate with other tools for automated certificate delivery. You can automate certificate processes and customize PKI workflows with ease. You can even integrate with third-party tools and applications, Mobile Device Management (MDM), Enterprise Data Management (EDM), Security Information and Event management (SIEM), and more.

WHY CHOOSE DIGICERT MANAGED PKI?

Simplified Certificate Management. Our Cloud Private PKI is integrated with our RESTful API, so you have access to free tools to simplify and automate certificate management.

Custom Certificate Profiles. Our team of expert engineers will advise you about what certificate profiles would be ideal for your organization.

Immediate Certificate Issuance. DigiCert certificates are issued within seconds without the threat of downtime or server outages.

Ready to Scale When You Are

Have questions specific to your organization's PKI needs? Talk to one of our PKI architects by calling 1.801.701.9690 or emailing enterprise@digicert.com.

Darin Andrew
Senior PKI Architect
darin.andrew@digicert.com

Darin Andrew is a Senior PKI Architect at DigiCert, where he applies a deep knowledge of public key infrastructure (PKI) and the DigiCert systems to help organizations effectively configure and optimize reliable, large-scale trust systems. Darin has contributed to many professional security-focused working groups, including DirectTrust, LXI Consortium, Society for Automotive Engineering (SAE), and others. Darin has consulted on PKI architecture for large and small cybersecurity needs.

digicert®