Journal of Science and Technology
ISSN: 2456-5660 Volume 7, Issue 06 (August 2022)
www.jst.org.in *DOI:https://doi.org/10.46243/jst.2022.v7.i06.pp20-30*

Enhanced Public Key Infrastructure for Public e-Services in India

Surya Teja Neerukattu

Department of Electronics and Communication Engineering,
Sathyabama Institute of Science and Technology, Chennai, India
neerukattusurya1@gmail.com

To Cite this Article
Surya Teja Neerukattu, **Enhanced Public Key Infrastructure for Public e-Services in India"** *",Journal of Science and Technology, Vol. 07, Issue 06,-August 2022, pp20-30*
Article Info
Received: 19-06-2022 *Revised:* 8-07-2022 *Accepted:* 13-08-2022 *Published:* 23-08-2022

Abstract

Public Key Infrastructure (PKI) is enabling infrastructure that provides authentication processes, public-key cryptography, and digital signatures to ensure confidentiality, data integrity, authentication and non-repudiation. PKI offers these "trust and security" services by using distinctive components like Certificate Authority, Encryption key management, end-user encryption software tools for storing, renewing, and revoking digital keys and secure socket layer certificates. This paper will present an architectural model to provide pki as a service in India for e-governance to provide public services in a secure way by addressing several shortcomings of the current infrastructure and its proposed extensions.

Keywords: Authentication, Encryption key management, Certificate Authority, Digital keys

Introduction

India is always an active adopter of new technologies thanks to its massive digital user base and adoptive business tech giants across sectors and industries. It holds 624.4 million active internet users (with 8.2% of increase per year) with an internet penetration rate of 45%. Being the world's largest democratic nation, governance per se has become complex and modern digital advances made increased public expectations from the government to adopt electronic public services.

To provide those digital public services, Government needs transparent and robust technologies and Stringent policies for wider inclusiveness and empowerment of all citizens and organizations. To reach these services to the public effectively agencies need to identify and authenticate end beneficiaries through secured encryption channels. These can be achieved by integrating PKI into application services [1].By realizing its potential, the Government of India appointed the Controller of Certifying Authorities (CCA) in 2000 by IT act which established root certifying authority of India. Governments need strong authentication, encryption, and digital signatures that are part of a comprehensive and scalable pki platform[2]. PKI is the basis on which governments can implement and monitor secure and trusted transactions whether between individuals and governments, businesses and governments or inter-state relationships, pki allows public entities to securely authenticate, and authorize all of its participants through the encrypted transactions. Even international cross border services can be leveraged from licensed certificate authorities of other countries in host country.

Current Status of PKI in INDIA

Government of India IT act 2000, empowers CCA to regulate certifying authorities in India. This central authority established root certifying authority to sign public keys of licensed certificate authorities in nation with its private key which are providing digital signatures based upon asymmetric cryptographic methods and certificates to end identities with online certificate status protocol services and also maintains an inventory .

Licensed CAs provides e-Sign services following e-authentication guidelines of government. Private keys of end users will securely store in hardware security modules which are maintained by third party entities belongs to respective CAs. Services provided by these certificate authorities included class 1,2&3 digital signature certificates ,e-sign, secure socket layer certificates, code signing certificates and time stamping services across different divisions. As per the conditions on which certificate authorities agreed with the central authority, they will provide respective services to certain sectors. For example, government authorities like Indian navy, army & air force need to provide these pki services to their own respective divisions.[3]

Related Implementation

As similar to above pki service model, government of USA in 1999 constituted non-hierarchical architectural Federal Bridge Certification Authority (FBCA) to provide interoperability in between several pki domains in country across different sectors and policies for issuing certificates comes in different levels like basic, medium hardware, High which depends upon the security provided to private keys by respective principle certificate authorities. Federal Bridge CA is not root CA, it only exchanges cross-certificate pair with each principal CAs which issue certificates having multiple object identifiers. These CAs are connected through network directory called "sneakernet" (not through direct network). [4]

Public programs like access certificates for electronic services (aces) are offering a channel for government agencies a method to issue basic level digital signing certificates for free which supports many implementations like electronic transactions and FBCA is ensuring to provide interoperability with CA services through these mechanisms similar to that IDABC pki Bridge Certification Authority was instituted by European commission to provide e-services for government officials and citizens. In India, IDRBT Certifying Authority which is specially dealt with Indian banking and Financial sector which will issue digital certificates.

Proposed Methods to Enhance PKI

This proposed architectural approach to improve PKI services in India is combination of methods which include management of keys through electronic key management, segregation of CA functionalities for both Internet and intranet purposes based on user load (including software implementation to provide certificate services to end-users, internal users) and finally, implementation for Distributed Denial-of-Service protection for pki infrastructure.

Generation and Management of Keys

Generating and controlling encrypted keys is vital part of any pki implementation because if these encryption keys got compromised, an unauthorized entities can decrypt and misuse these keys. An ideal key management system need to include generation, storage, destruction, replacement of keys. In India, generally there are four types of key management systems are in use depending on the requirement. They are: an HSM module, key management virtual appliance (mostly dedicated server), key management software, cloud key management servers and key management SaaS from vendors. For public services it always advisable to use hardware module to secure keys. Fig1 is the architectural model implementation of HSM for managing keys. [5] General motive is to use the encryption keys which are generated on the key managing system protected by hardware module should only consumed by the application server during runtime or for running at particular service on that server.

.

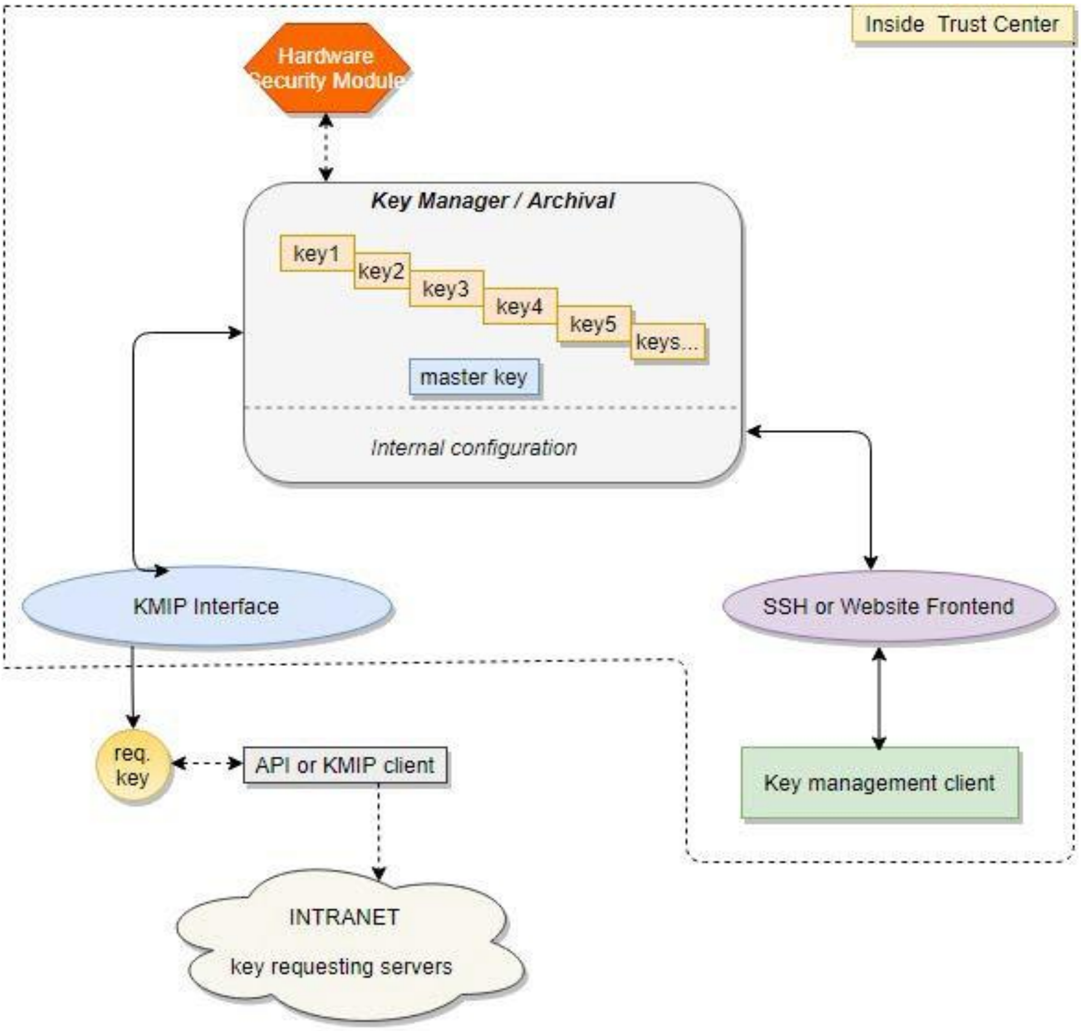


Figure 1: Architecture of Key Management

Keys are not stored with the consumer server .end-user server can get encrypted keys after authentication and transmission of these keys only allowed to secure channel. Internal configuration section maintains the access policies and server authentication setup. For wide range of encryption needs, a key management appliance needs to cater heterogeneous key Management (symmetric and asymmetric) and also need to leverage automated operations like key expiry and key rotation tasks through policy driven actions. Inside Government trust center, administrative role restrictions are needed according to their scope of responsibilities which can be achieved by allowing LDAP or active directory authentication for user key access for application servers. An alerting system can also be deployed through SNMP (port 161).Intelligent key sharing can engender through active- active clustering mode.

User keys are managed by encrypted master key which will be integrated with hardware security module. Master key will be generated automatically by making partitioning HSM through vendor specific methods, for example making partition using special HSM specific keys (using pin entry devices physically). For restoring back of the main HSM we can use remote backup HSM (can use PED devices or password authentication) by using password backup file (master key will not be included). For respective ministerial trust centers, Key management client though secure shell (through serial cable or secure shell client) or website will provide administrative interface. For web interface, browser preferably need to use https protocol through default port of 9443. For end user interface, we can take advantage of KMIP which is open protocol by OASIS, but user application need to authenticate using client certificate. Alternatively, there will be vendor specific APIs which will use network attached encryption protocol (majorly used for bring your own key requirements).

Open source software: HashiCorp Vault key management, Egnyte software for cloud related key management

4.1. Segregation of CA's Intranet and Internet Requirement

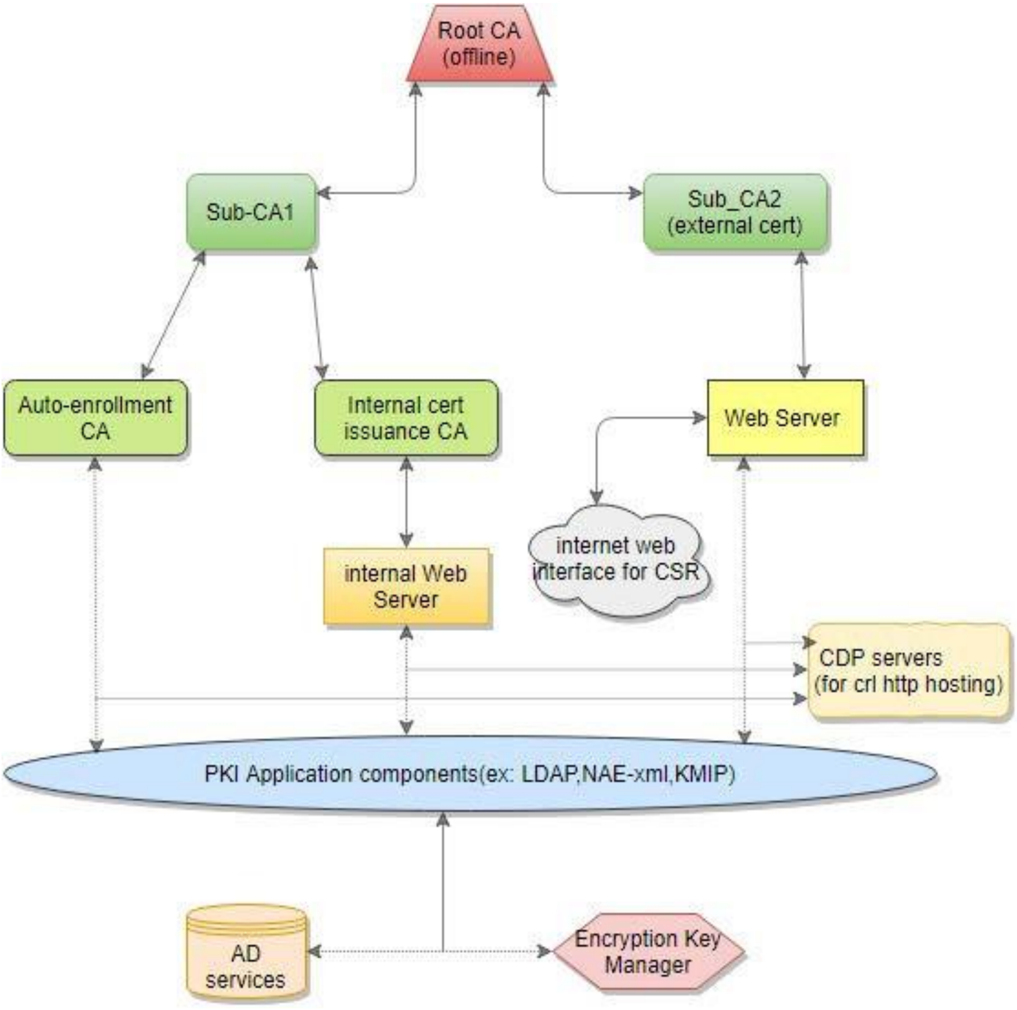

Root CA (offline)
Sub-CA1
Sub_CA2 (external cert)
Auto-enrollment CA
Internal cert issuance CA
Web Server
internet web interface for CSR
internal Web Server
CDP servers (for crl http hosting)
PKI Application components(ex: LDAP,NAE-xml,KMIP)
AD services
Encryption Key Manager

Figure 2: Functional Representation of Subordinate CAs

For each ministry of government depending upon the requirement of certificates need to issue for its department or to citizens, It is always efficient to separate root certificate authority work to subordinate intermediate CAs for both intranet and internet to reduce the bottleneck on root CA and to make it offline for limiting the surface area so that respective ministry will have advantage in case of subCA being compromised to bring root CA online to stop its operations and revoke certificates which are issued by that sub intermediate CA. but before making root CA offline, it needs to be standalone CA as it was coupled with the active directory of respective department network. Root Ca's only pertinent functions are creating periodic certificate revocation list files and to sign and revoking subCA's certificate signing requests. [6] As e-government services are expanding through-out the nation. It will increase the end users, internal Gov. employees and their working systems which are needed to be included in the government department. It will become very difficult to manage one or two CAs to handle both external and internal certificate services and small misconfiguration can lead to whole pki infrastructure compromise due to high surface area. So, In proposed system there are two CA for handling internal and external services simultaneously.

4.2.1. Implementation of Auto-Enrollment for Intranet Devices:

It is very difficult to manually configuring client authentication device certificate to each government device in internal network. So, it is evidently better to automatically pushing device certificate to device for enabling authentication. Inside Government trust center, all devices such as laptops, mobiles, tablets, PCs, e.t.c, which belongs to Active directory of the government organization needs to connect to the internal WLAN authenticated through Subordinate CA (auto enrollment CA issued device certificates (with client authentication enabled). [7]

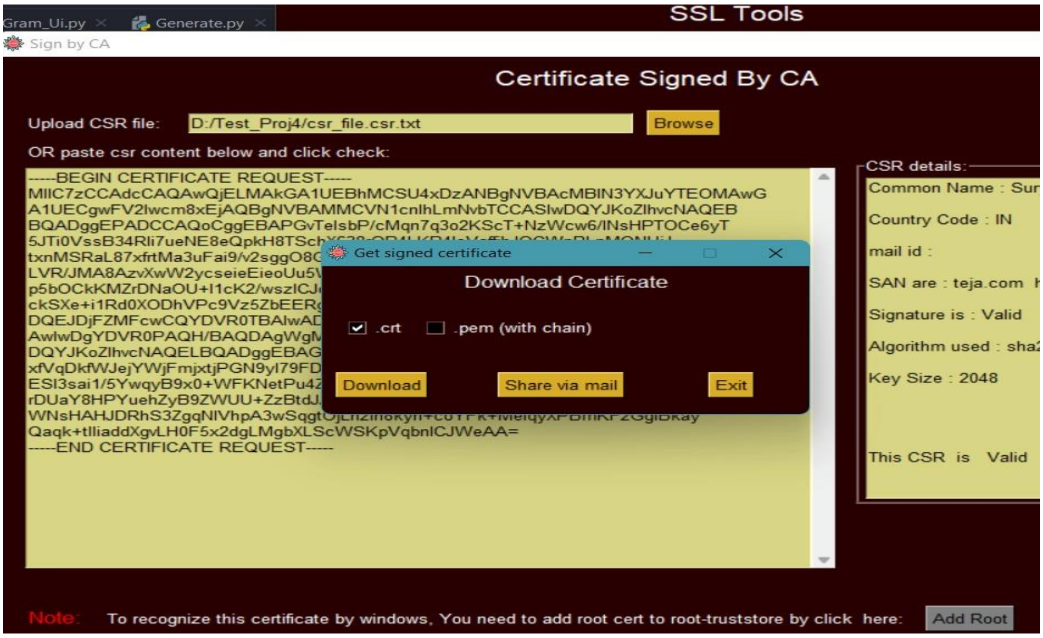
When new device join to government org. domain active directory provides user and device specific info (gives username & password authentication) and CA issued certificate will be linked to account of that device in AD. Then Device Management platform like Cisco Meraki or MobileIron to directly manage security and configure settings in device (enables notifications through active sync bridge) and it requests device certificate and installs into the device after receiving it from CA. Auto-enrollment CA checks the attributes of certificate request and pushes client auth. certificate to device certificate manager to enable authentication. Certificate will expired based on its expiry date and it will reflect on CRL, device management will remove the certificate and will request new certificate from CA.

4.2.2. Certificate Issuance

Internal certificate issuance CA and Sub_CA2 (for handling Citizen certificate requirement both will issue SHA2 secure socket layer certificates, WLAN certificates, server and client authentication (support both RSA and EC templates. For using RSA encryption algorithm, certificate signing request need to be generated with minimum key size of 1024 bits whereas to use advanced EC encryption, Certificate signing request need to be generated with minimum of 384 bit key size. And both internal and external web servers use RASQL (for example, MS SQL database to store to store policies, bulk user data, logs and reports. Active directory will have entry of every user account of each server, Certificate revocation lists from CDP points, CA certificates and security policies using LDAP compliant directory service. All CDP locations (in CA server and http location) certificate revocation lists need to publish periodically and they will updated to the main CDP server .In critical infrastructure deployments, It is advisable to maintain two HSMs as active and passive. Whenever there is configuration change for HSMs, we can take backup using remote backup HSM. If active HSM failure happens, passive HSM need to replace it. [8]

Open-source software: Openssl to create,sign and revoking certificates, OpenLDAP to store certificates and CRLs .OCSPD can be used to verify status of certificate. OpenCA is popular opensource implementation of certificate authority based on apache and optional apache modssl, and its best alternative is EJBCA which work through j2ee and platform independent. OpenXPKI which is enterprise-grade Trustcenter software supports third party softwares and can us HSM through it. [9]

4.2.3. Software Implementation



Certificate Signed By CA

Upload CSR file: D:/Test_Proj4/csr_file.csr.txt

OR paste csr content below and click check:

```
-----BEGIN CERTIFICATE REQUEST-----
MIIC7zCCAdcCAQAwQjELMAkGA1UEBhMCSU4xDzANBgNVBAcMBIN3YXJuYTEOMAwG
A1UECgwFV2lwcm8xEjAQBgNVBAMMCVN1cnlhLmNvbTCCASIwDQYJKoZIhvcNAQEB
BQADggEPADCCAQoCggEBAPGvTelsbP/cMqn7q3o2KScT+NzWcw6/INsHPTOCe6yT
5JTi0VssB34Rli7ueNE8eQpkH8TSch
txnMSRaL87xfrtMa3uFai9/v2sggO8C
LVR/JMA8AzvXwW2ycseieEieoUu5V
p5bOCkKMZrDNaOU+I1cK2/wszICJ
ckSXe+i1Rd0XODhVPc9Vz5ZbEER
DQEJDjFZMFcwCQYDVR0TBAIwAD
AwIwDgYDVR0PAQH/BAQDAgWgM
DQYJKoZIhvcNAQELBQADggEBAG
xfVqDkfWJejYWjFmjxtjPGN9yI79FD
ESI3sai1/5YwqyB9x0+WFKNetPu4Z
rDUaY8HPYuehZyB9ZWUU+ZzBtdJ
WNsHAHJDRhS3ZgqNIVhpA3wSqgt
Qaqk+tIliaddXgvLH0F5x2dgLMgbXLScWSKpVqbnICJWeAA=
-----END CERTIFICATE REQUEST-----
```

Download Certificate

☑ .crt ☐ .pem (with chain)

CSR details:
Common Name : Sur
Country Code : IN
mail id :
SAN are : teja.com
Signature is : Valid
Algorithm used : sha
Key Size : 2048

This CSR is Valid

Note: To recognize this certificate by windows, You need to add root cert to root-truststore by click here:

Figure 3: Implementation of certificate Authority and its services

Scripting language used: python (interpreter version 3.10), Power shell scripting

Graphical user Interface: PYGUI and tkinter

Operating system tested on: Windows

Libraries used: pyCryptodome, Openssl, serialization (crypto package), SMTP

Features tested [9.g]: keys generation, CSR validation, requesting signature from CA, self-signed certificate, issuing through soft-token (in pkcs#12),*file conversion, certificate installation

4.2. Distributed Denial-of-Service Protection

As user load on certificate authorities is huge due to number of different services offered by that pki infrastructure via internet is increasing, DDos (Distributed Denial-of-Service) protection is key element of overall security and connectivity measures helping to guarantee a needed level of internet connection's availability to provide public services. In intranet services of pki infrastructure, DDoS attacks are mitigated before they could do any harm, so bandwidth is preserved and network infrastructure devices are not overloaded by high volume of forwarded data. Hence, It is imperative for internet facing applications where DDoS protection is need to be implemented (like traffic statistics collection and routing manipulations). [10]

Currently, There are two ways to provide DDoS protection:

a) DDoS protection services-independent provider, In this DDoS protection is provided by single provider with high performance scrubbing centers.But problem with this is to need to collect and export statistics data about internet traffic or need manual intervention.

b) DDos protection by ISP - a simple setup for DDoS detection in which protection is provided by ISP for a particular location. And it differs by service to service inside pki infrastructure. Drawback with this implementation is that some ISPs not support DDoS protection and non-scalable. As public schemes services concerns, Service- independent provider for DDoS protection is best as it has scored well in all evaluation criteria.

In this implementation, Internet access link with fixed bandwidth. Depletion of all available bandwidth on the link will make on-site located services inaccessible from internet and trust center users will not be able to use SaaS in internet and in Border firewall where security policies are applied and NAT from internal IP addresses to public IP is performed. No internal Ips and no unencrypted sensitive data can appear beyond this point. Under normal conditions there will be as usual traffic to/from trust center to SaaS services but rerouted traffic occurs only under active DDoS attack in case of using ddos protection service. High volume traffic, which is

now mix of legitimate and malicious communication is redirected using routing manipulation technique to scrubbing center for mitigation and form there legitimate traffic (DDoS traffic removed in scrubbing center sent to trust center over secure tunnel(Ip/GRE)

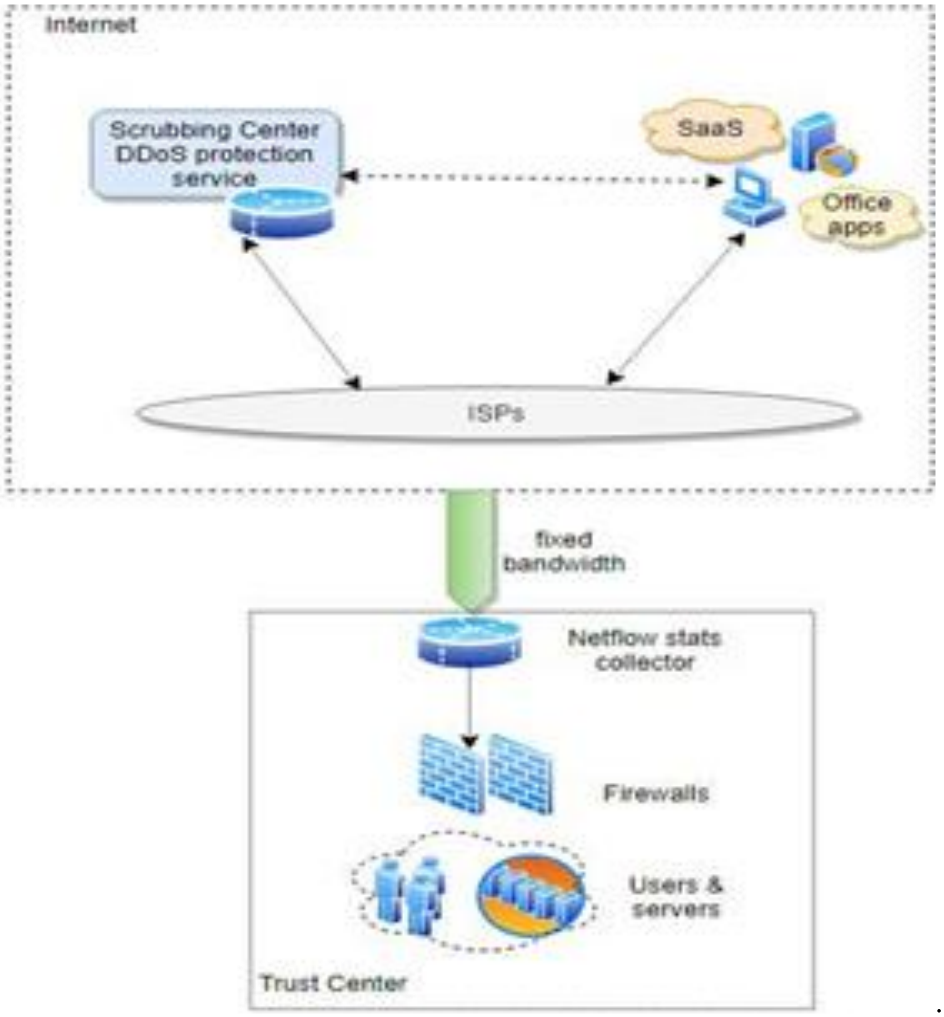


Figure 4: Distributed Denial-of-Service Protection Architecture

Netflow stat collector is needed to collect traffic statistics on internet-facing applications, there will be no private IP addresses nor unencrypted sensitive data in this part of the network [11]. Export of traffic stats to scrubbing center by analyzing anomalies indicating DDoS attack,They will be shared unencrypted from trust centers to service providers device in closest scrubbing center.

Conclusion

This paper has proposed and reviewed a range of infrastructural, management, and operational enhancements to current pki implementations in India for providing public services to citizens by minimizing the risk of key compromise. The advantage of these methods is to mitigate problems with conventional pki such as single point of failures and server outages. The certificate issuing tool was implemented and made available on Github under free MIT license. For future work, Back up and restoration infrastructure need to be there for overall pki not for just key management, there is need of better external user management and also firewall and proxy servers need to be deployed as per service requirement. Electronic key management connector, RASQL services need to be introduced in implementation software and need to include more file conversions such as java key store.

References

[1] B. Klicek and D. Plantak Vukovac, “Information society and e- Government developments in Croatia,” Informatica 31, 2007, pp 367-372.

[2] https://cca.gov.in/pki_framework.html , last access date 17.06.2022.

[3] R.Housley, T. Polk “Planning for PKI: Best Practices Guide for Deploying Public Key Infrastructure”, Wiley; 1 edition, 2001.

[4] Barbara Lörincz and group of authors, “Digitizing Public Services in Europe: Putting ambition into action,” 9th Benchmark Measurement, European Comission, Directorate General for Information Society and Media, December 2010.

[5] Yazdanpanah S, Saman SC, Mazdak Z, Reza K. Security features comparison of master key and IKMcryptographic key management for researchers and developers. In: International Conference on SoftwareTechnology and Engineering; 2011:365-369

[6] E. Winjum and A. Fongen, “Model and specification for analyzing the scalability of a public key infrastructure,” Norwegian Defence Research Establishment, Tech. Rep. 2009/01546, 2009

[7] Harn L, Hsu C, Xia Z. Lightweight group key distribution schemes based on pre-shared pairwise keys. IETCommun. 2020;14(13):2162- 2165.

[8] R. Vacca, “Public Key Infrastructure: Building Trusted Applications and Web Services”, Auerbach Publications, 1 edition, 2004.

[9] a. Openssl, https://www.openssl.org/

b. EJBCA, http://www.ejbca.org/index.html.

c. OpenCA, http://www.openca.org/

d. OCSPD, https://www.openca.org/projects/ocspd/

e. Apache mod_ssl module, https://httpd.apache.org/docs/current/mod/ mod_ssl.html

f. OpenXpki, https://www.openxpki.org/

g. SSL toolkit application, https://github.com/SuryaTeja-N/SSL-Tools

[10] Levy-Abegnoli, G. Van de Velde, C. Popoviciu and J. Mohacsi, “IPv6Router Advertisement Guard”, RFC6105, Internet Engineering TaskForce, February 2011

[11] Ahmad Sanmorino and Setiadi Yazid, “DDoS Attack Detection Methodand Mitigation Using Pattern of the Flow”, ICoICT’13, InternationalConference of Information and Communication