Global Information Assurance Certification Paper

Modeling and Simulation of Public Key Infrastructure Applications

Kelley R. Klepzig

Abstract

Public Key Infrastructure (PKI) is one of the primary weapons in the Department of Defense's (DoD) "defense in depth" information assurance strategy, and will be used to sign all DoD e-mail messages by October 2003. PKI uses asymmetric "public" and "private" keys to sign and/or encrypt e-mail messages and other information. The effect of this additional workload on the existing network and servers was investigated using OPNET Capture Agent, Application Characterization Environment (ACE), and Modeler. Another application which uses PKI, the Secure Single Sign-on prototype, was also investigated using the OPNET Capture Agent, ACE, and Modeler. These types of investigations allow the manager to ascertain the true benefits and performance costs in implementing PKI.

Introduction

Achieving Information Superiority in the highly interconnected, interdependent, shared-risk DoD environment requires that the Department's Information Assurance (IA) capabilities be applied within a management framework that considers the pervasiveness of information as a vital aspect of warfighting and business operations. The technical strategy that underlies DoD IA is Defense in Depth, in which layers of defense are used to achieve our security objectives. The DoD PKI is a supporting layer of this strategy, providing a vital element for a secure IA posture for the Defense Information Infrastructure (DII).

Commercial businesses also employ this Defense in Depth strategy, although perhaps not by this name. Mike Bobbit, in "Regardless of its depth and complexity, PKI is still only one piece of any organization's security infrastructure. For that reason, a PKI trust policy needs to stand on its own and mesh with an organization's overall security policies. Glaring discrepancies between the two policies may render one or both invalid."

The DoD PKI strategy recognizes that a traditional, Government-developed implementation will not be able to keep pace with a strategy based on commercial technology and services. It recognizes that the DoD PKI must employ an incremental, evolutionary approach using open standards, based on commercially available products and services that can keep pace with the technology rollover and constantly evolving applications and standards inherent in the Information Technology (IT) environment. With that, it must still maintain appropriate levels of security, embracing secure interoperability both within the DoD and externally with Federal and international counterparts and with business partners.

Implementation of PKI is a significant undertaking, and commercial businesses are finding that it is worth the effort. Brink, Derek, in "PKI and Financial Return on

Investment," PKI Forum, August 2002, URL: http://www.pkiforum.org/pdfs/Financial_Return_on_Investment.pdf, concludes "the total cost of ownership for implementing an enabling e-security infrastructure such as PKI is significantly less than the financial returns made possible by PKI-enabled applications, when revenues, costs, compliance and risks are understood and quantified."

The individuals, programs, and systems that carry out or support the broad range of missions and operations of the DoD perform a variety of activities. These diverse activities represent an ever-expanding need for IA capabilities in DoD operations. Traditionally, DoD has satisfied these needs with stand-alone cryptographic components. In today's IT-rich environment, DoD's IA needs are being addressed with security features integrated into the many communications and information processing system components that comprise the DII. PK technology is rapidly becoming the technology of choice to enable security services within these systems. These security services include: identification and authentication; data integrity; confidentiality of information and transactions; and non-repudiation to facilitate mission-related and eBusiness transactions internal to the Department and with external organizations.

PKI, as defined herein, refers to the framework and services that provide for the generation, production, distribution, control, revocation, recovery, and tracking of PK certificates and their corresponding private keys. The DoD PKI will support registration of users, dissemination of certificates, and a full range of certificate management services. This provides the critically needed support to individuals, applications, and network devices that provide secure encryption and authentication of network transactions as well as data integrity and non-repudiation.

Certificates are instruments used to convey trust. The DoD PKI will provide three types of certificates: identity certificates (used for authenticated access), e-mail signature, and key establishment (confidentiality) certificates. There are profiles within these types that will support certificates for servers, e-mail signature services, and e-mail confidentiality services. To achieve common certificates across the entire DoD, the DoD PKI identity, e-mail signing, server (device), and encryption certificates will have a minimum/common set of attributes as specified in the certificate profile section of the DoD X.509 CP. As the PKI evolves, it is possible that additional certificate types will have to be provided. Other types of certificates such as network access and object-signing certificates will be supported by the PKI as operational requirements dictate.

Public key technology provides the ability to conduct electronic transactions in a secure fashion, providing a means for the following:

- Authentication – Mechanisms to strongly authenticate user identities;
- Confidentiality – Ability to enable strong encryption that can protect the privacy of information transferred during a transaction;
- Integrity – Capability to ensure that transactions have not been modified by an unauthorized party;

- Key recovery – Capability for authorized users to obtain cryptographic keys needed to recover information protected with encryption keys that may have been lost or damaged; and
- Non-repudiation – Ability to validate that specific users were involved in a transaction.

Public key technology provides the mechanisms to implement these security services, enabling the broad scope of business process re-engineering activities that will lead the DoD to a paperless environment. These services are available to individuals, network servers, network devices (e.g., routers and gateways), and properly configured software applications.

This paper will examine three aspects of DoD PKI and its applications. E-mail, directory system, and secure single sign-on (S3) will be examined. OPNET modeling and simulation will be used to investigate aspects of each of these applications.

E-Mail

Traditional symmetric cryptography uses a single key to both encrypt and decrypt information. Public key technology is based on asymmetric key-pairs. A key is an electronic file, and a pair of keys is created at the same time by a special software program. The keys are not identical, but have a mathematical relationship so that they will only work with each other to encrypt and decrypt information. Information encrypted with one key can only be decrypted by the other, and vice versa. Figure 1 shows this concept for e-mail, where the originator uses the recipient's public key to encrypt a message, and only the intended recipient can decrypt the message, since he is the only one with his private key needed to decrypt it.

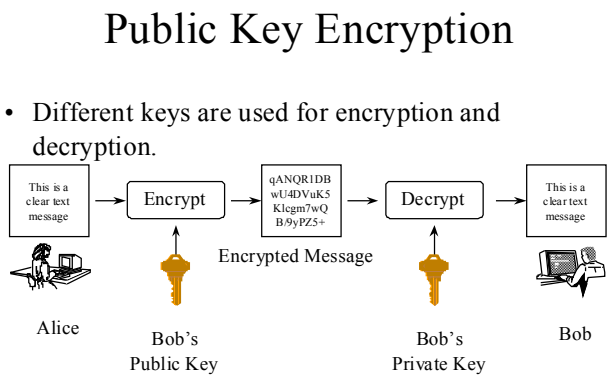


Figure 1: PK Encryption

Since asymmetric public key encryption is less efficient than traditional symmetric encryption in terms of file size, a combination of public key encryption and symmetric encryption is used, as shown in Figure 2. Symmetric encryption is used to encrypt the message, and public key encryption is used to encrypt the symmetric key. This

provides the advantages of public key encryption without the large file size that would otherwise result.

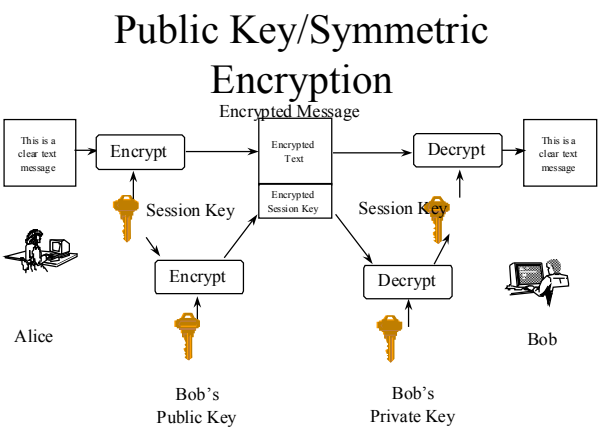


Figure 2: PK/Symmetric Encryption

Public key encryption is used when signing an e-mail message as well as in encrypting it. This process is shown in Figure 3.

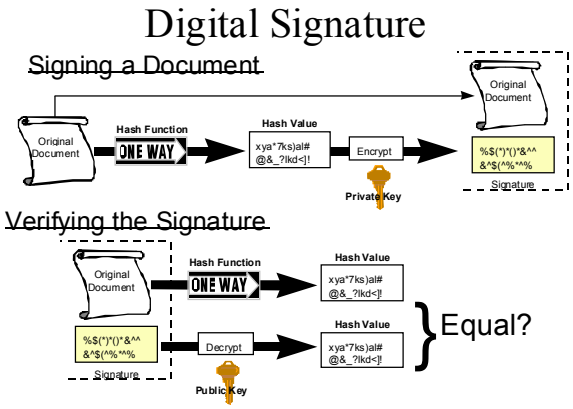
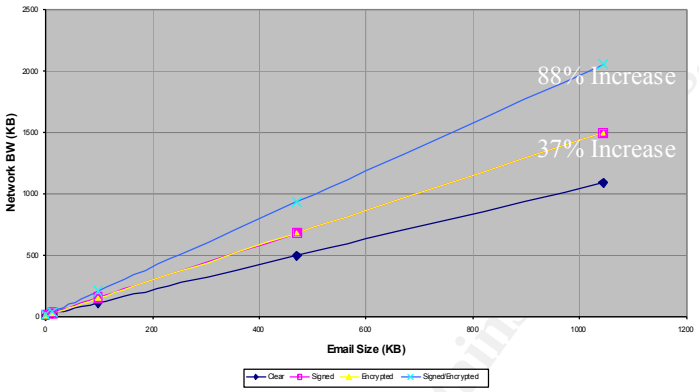


Figure 3: Digital Signature

In signing a document, the original document is subjected to a one-way hash function, and the result is encrypted with the originator's private key. This "signature" is transmitted along with the original unencrypted document. The recipient decrypts the "signature" with the originator's public key, and also subjects the received document to the same one-way hash function that the originator used. If the results of these two operations are identical, the recipient knows that the message was truly sent by the claimed originator (since he was able to decrypt the "signature" with the originator's public key) and that the content of the message was not tampered with during transmission (since the hash values matched). When a message is signed, the originator's public certificates are appended to the message.

The signing and encrypting operations increase the size of the message. Measurements were made of the size of a basic message which was unsigned and unencrypted, signed only, encrypted only, and signed and encrypted. This was repeated with several different size messages, and the results are presented in Figure 4.



Client to server traffic based on email size

Figure 4: Message Size Growth with PKI

The growth is significant, up to 88 percent when a large message is both signed and encrypted, and 37 percent when the message is either signed or encrypted.

In order to model these e-mail transactions, they were captured and placed in individual files. This was done with the OPNET Capture Agent software. This captures transactions much like a conventional sniffer. The captured file was then imported into the OPNET Application Characterization Environment (ACE) module. One of the functions available in ACE is the Application Message Chart, which depicts transactions between selected devices. Figures 5 and 6 show this chart for a message with a 1400KB attachment, both unsigned and unencrypted (Figure 5) and signed and encrypted (Figure 6). The charts are color coded, with yellow being less than 500 bytes and blue being greater than 1460 bytes. It is easy to see that the signed and encrypted transactions result in much more traffic.

ACE depiction of 1400KB attachment transaction
Unsigned and Unencrypted

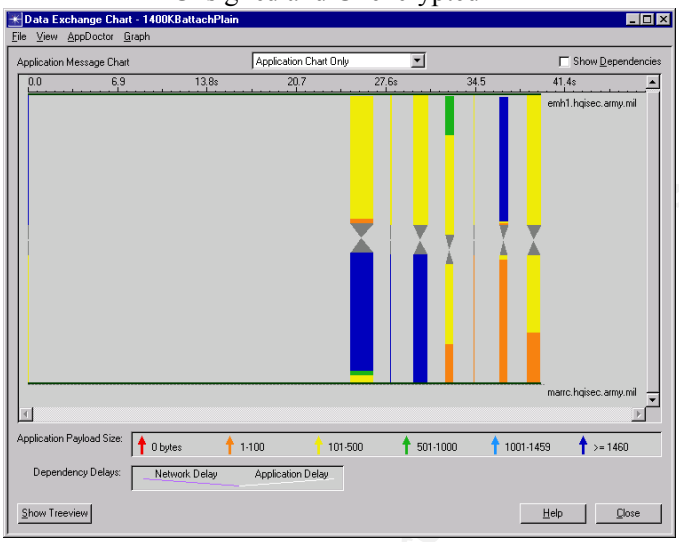


Figure 5: ACE Chart, Unsigned and Unencrypted

ACE depiction of 1400KB attachment transaction
Signed and Encrypted

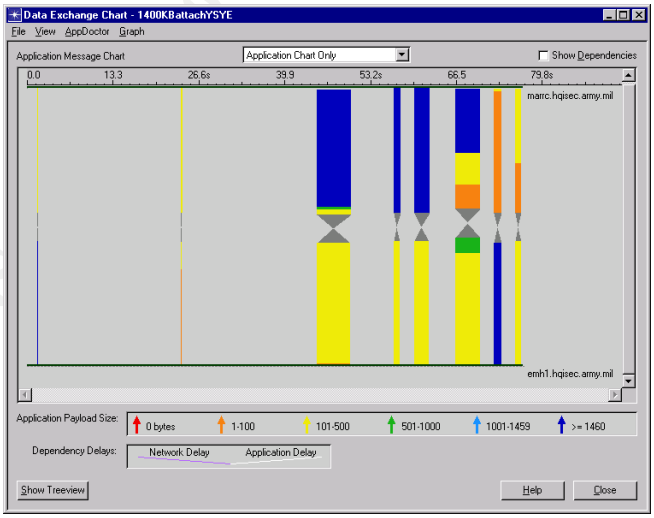


Figure 6: ACE Chart, Signed and Encrypted

ACE files of typical email transactions were imported into OPNET Modeler, and the program was used to configure the network from the ACE file. This network consisted of the single client and the server used in collecting the data. The client was then

replicated 100 times to form a network with one server and 100 clients as shown in Figure 7.

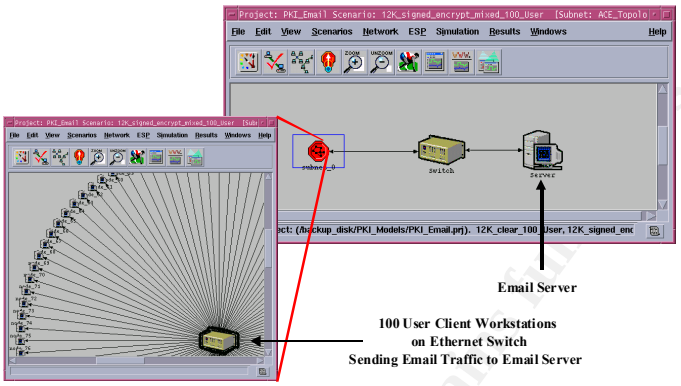


Figure 8: Email Simulation Topology

The ACE files were used as the input for each client and the response from the server. Therefore the actual email traffic was used in the model, rather than some estimated traffic. Several simulation runs were made to compare various parameters, both for the unsigned and unencrypted traffic and the signed and encrypted traffic. Figure 9 shows the effect on bandwidth and response time. The results compare all 100 users sending unsigned and unencrypted traffic with 50 of the users sending signed and unencrypted traffic and the other 50 sending signed and encrypted traffic.

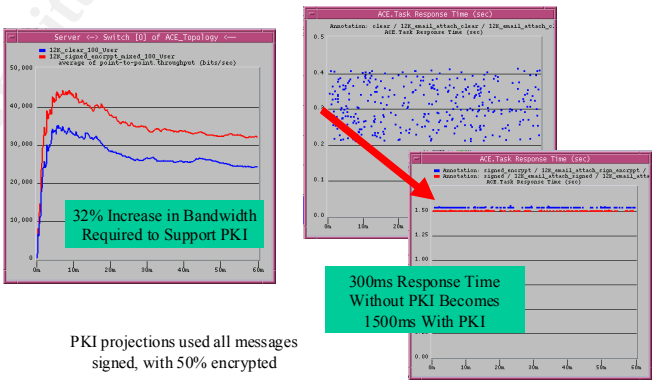


Figure 9: Performance Effects of PKI – Simulation Results

Additional email simulations will be run after data concerning actual and forecast user populations, message size distributions, bandwidth, and background traffic are gathered

and incorporated into the model. The model can also be utilized to compare centralized versus distributed email server configurations.

PKI Directory System

A directory is used as a repository for the distribution of the subscriber certificates and Certificate Revocation Lists. Appropriately, Certificate Revocation Lists contain the identities of the certificates that have been revoked, and therefore should not be used. In addition to distribution of certificate management information, directories can be used to distribute other subscriber information such as e-mail address, phone number, and postal address. The Defense Information Systems Agency (DISA) maintains PKI directory servers at Chambersburg, PA and Denver, CO. These two servers are mirror images of each other. As noted above, a message originator needs the public certificate of the recipient, if the originator is to encrypt the message for the recipient. There are several ways to obtain the certificate, but the directory is one of them.

In addition to maintaining a centralized directory server, it is possible to shadow the information down to regional directory servers, and from there information of local interest can be shadowed down to local servers at each site. The directory information would be mastered only at the centralized server, but the directory information most used by subscribers at each site would be locally available at the site. This topology was modeled with OPNET. In the previous e-mail example, the network was initially constructed by OPNET using input from ACE. In this directory example, the network was manually constructed in accordance with what might be a typical network. This network is shown in Figure 10.

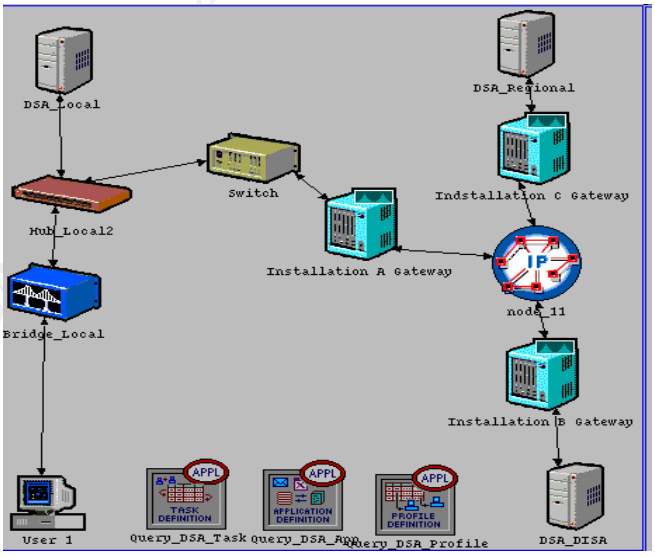


Figure 10: Directory Server Configuration

The network to the left of the IP cloud in Figure 10 represents the elements at a local site, with one user shown. The network was expanded so there were 10 sites and 76 users at each site, as shown in Figure 11.

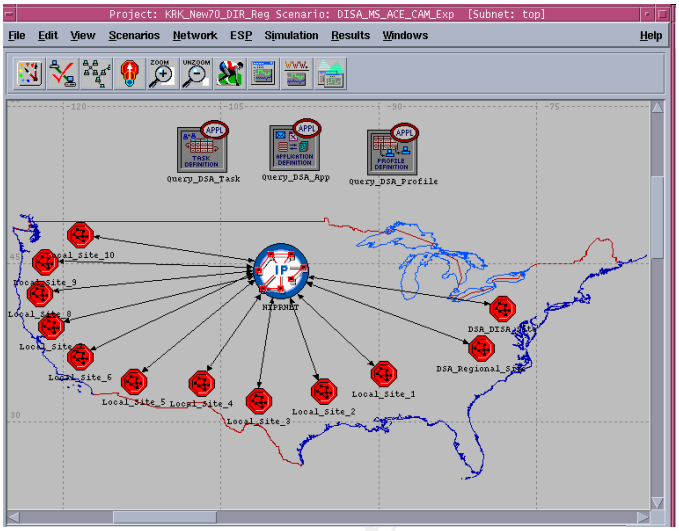
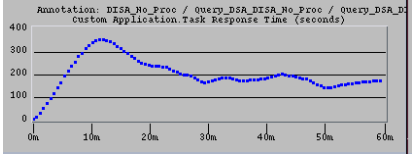


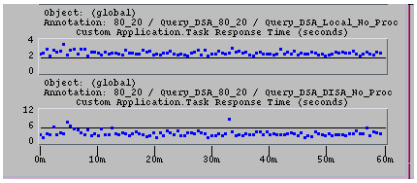
Figure 11: Directory Server Network

The traffic from each user simulated a request for another user's certificate, with the request specified to be 100 bytes long. The response from the directory server, simulating the other user's certificate, was specified to be 1000 bytes long. Several simulations were run, with one to 760 users making a single request, to continuous requests exponentially distributed. The requests were first made to only the single centralized server. The model was then reconfigured so that 80 percent of the requests went to the local server and 20 percent of the requests went to the centralized server. The results of the simulations are shown in Figure 12. When a single centralized server attempted to serve all users, the delay experienced by the users built up to over 300 seconds, and settled down to about 200 seconds. When the load was distributed 80-20 percent over local and centralized servers, the delay experienced by requests to the local directory was approximately three seconds, while the delay experienced by the requests to the centralized directory was approximately four seconds. The comparison of delays is shown in Figure 12. Note that, while the model used the same characteristics for all servers, it was an unrealistically slow server, and the request/response transactions were assumed rather than actual transactions. Therefore the results are not necessarily real-world, and are intended only to show evaluation techniques of modeling and simulation.

100% Centralized



80% Local,
20% Centralized



NOTE: Results are based on unrealistic assumptions (extremely s low servers, assumed request/response sizes, etc), and are intended only to show evaluation techniques of modeling and simulation.

Figure 12: Local/Centralized Directory Results

The above results were based on the 100-byte request and 1000-byte certificate which were specified to the model. A sniffer was then used to capture transactions between a user and a directory server, with the user requesting another user's certificate, and the directory server responding with that certificate. This captured file was input into ACE, and the resultant ACE file was used as the user/directory server traffic rather than the 100/1000-byte traffic. This "live" captured traffic contained a large amount of human "think" time. When the simulation with captured traffic was first run, the "think" time was interpreted by OPNET as required processing time, and the simulation ran for hours without productive results. The attribute for the server processor was then changed to "Contention Already Modeled". This was appropriate, since the human "think" time was not contending for processor time, and should not be counted that way. The results of the simulations with the captured traffic are reasonable, but not readily displayable.

Single Secure Sign-on (S3)

Single Secure Sign-on is a very convenient tool in that it allows a user to sign on to a single web site and, from that site get access to any other web sites to which he is authorized. The Army Materiel Command has a prototype S3 system which uses a proxy server model. The user signs on to the proxy server, and the proxy server signs on to the backend server. The user is then allowed access to the backend server, assuming he is authorized to do so. The access control list is still maintained by the backend server. Figure 13 shows the workflow with and without the S3 server. Without the S3 server, the user goes directly to the backend server, but has to separately sign in to each different backend server. With the S3 server, both the request and response go through the S3 server, but the user only has to sign in once, to the S3 server.

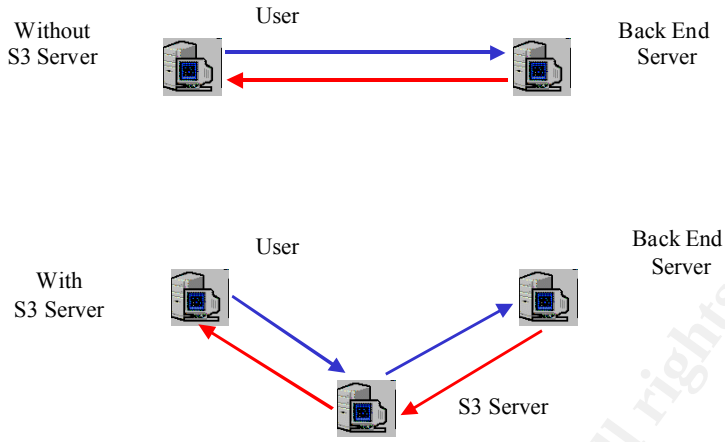


Figure 13: Workflow, with and without S3

Data files were captured from these transactions and imported into ACE. OPNET was used to construct the network, and simulations were run on the model network. The user was then replicated 100 times to form a network with one S3 server and 100 users as shown in Figure 14.

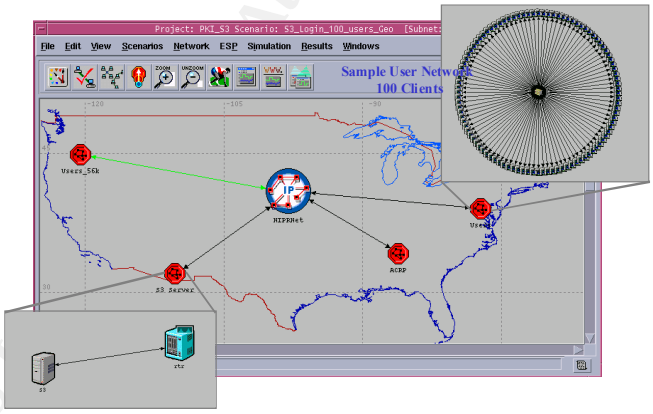


Figure 14: S3 Topology

Simulations were run to compare the two configurations, with and without the S3 server. With the S3 server in the configuration, the login process took less time, but the actual transaction response time increased form 400 ms to 850 ms. This is shown in Figure 15.

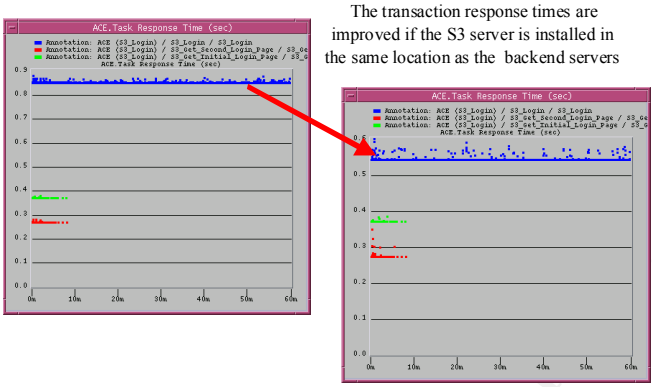


Figure 15: S3 Response Time

The configuration was then changed so that the S3 server was co-located with the backend server. Therefore the traffic from the S3 server to and from the backend server traversed the local Local Area Network rather than the slower transmission cloud. As a result, the response time decreased from 850 ms to 550ms, as shown in Figure 16.

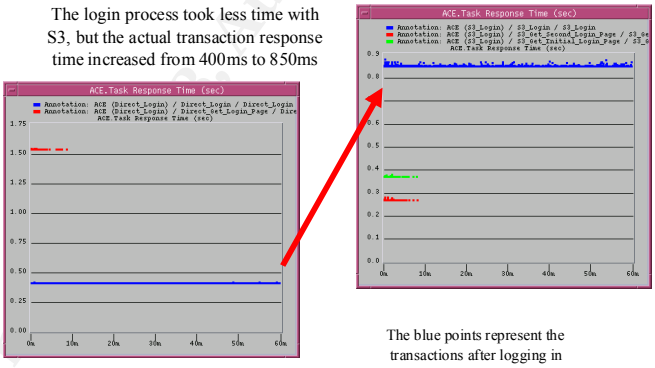
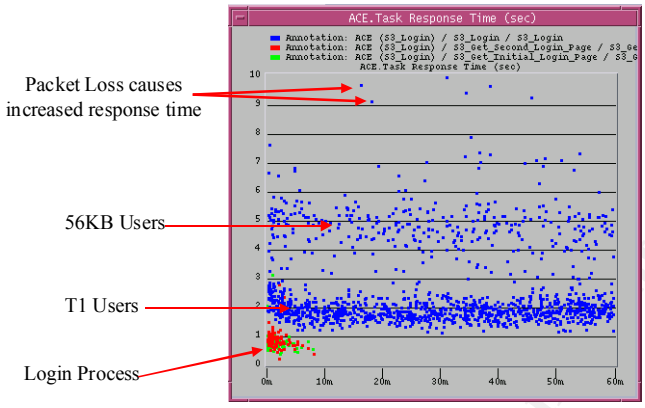


Figure 16: S3 Response Time, Co-located Server

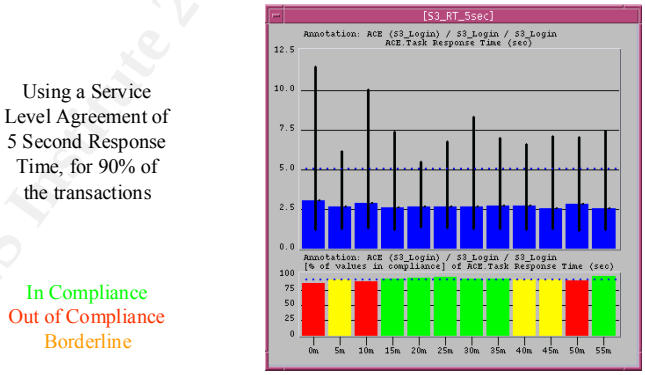
The configuration was then changed to incorporate other aspects that might be found in a real network. A 56Kbps user was added, and network factors of variable latency, packet loss, and background traffic, were added in the network. As shown in Figure 17, the 56Kbps users experienced increased response time, and the packet loss caused significantly increased response times.



Background traffic, variable latency in NIPRNET,
packet loss, and 56KB user connections are included

Figure 17: S3 Response Time, Increased Fidelity

For this same configuration, a service level agreement was set, requiring a 5-second response time for 90 percent of the transactions. Five-minute measurement periods were selected. While it is impossible to readily tell if this criterion is met from the data of Figure 17, it is obvious from the OPNET Expert Service Prediction (ESP) module display of Figure 18. The green/yellow/red color codes for met/borderline/unmet criterion makes it very east to read.



Same results as previous graph – displayed differently

Figure 18: Service Level Management

Conclusion

This paper has shown how OPNET modeling and simulation can be used to provide answers to real-world PKI questions. The results presented are based on assumed parameters which might not reflect real-world values; therefore the results might not reflect real-world results. As the assumptions made about the parameters can be replaced with actual values, the results will also come closer to actual values. These types of results can be used to make management decisions based on the relative benefits and costs of a particular PKI implementation.

References

[1] Bobbit, Mike, "PKI Policy Pitfalls." TrueSecure Corporation's Information Security. July 2001, URL: http://www.infosecuritymag.com/articles/july01/features_pki.shtml (4 Apr. 2003)

[2] Brink, Derek, "PKI and Financial Return on Investment." PKI Forum. August 2002, URL: http://www.pkiforum.org/pdfs/Financial_Return_on_Investment.pdf (4 Apr. 2003)

[3] Public Key Infrastructure Roadmap for the Department of Defense, Version 5.0, DoD Public Key Infrastructure Program Management Office, December 18, 2000

[4] Public Key Infrastructure Implementation Plan for the Department of Defense, Version 3.1, Public Key Infrastructure Program Management Office, December 18, 2000

[5] Modeling and Simulation Techniques for Comparing PKI Directory Architectures in a Military Environment, Ralph Meacham and Kelley Klepzig, OPNETWORK 2000, August 29, 2000

[6] U.S. Army Materiel Command Class 3 PKI for Individual Messaging Technical Assessment, Cost Estimate, Design Outline, U.S. Army Information Systems Engineering Command, July 2001