

Wireless PKI and Distributed IDS for Securing Intranets and M-Commerce

Kai Hwang

Internet Security and Pervasive Computing Laboratory
University of Southern California
Los Angeles, CA. 90089-2560 USA

Keynote Address

IEEE Third International Conference on Parallel and Distributed
Computing, Applications, and Technologies (PDCAT 2002)
Kanazawa, Japan, September 4-6, 2002

Abstract:

Recent R/D advances are presented in this keynote address on wireless and security technologies. To access Internet from mobile devices, the existing *public key infrastructure* (PKI) must be modified to work with limited wireless network bandwidth and low computing and memory capacity of handheld devices. A complete security chain is needed from smart cards to mobile clients, *wireless PKI* (WPKI) platform, and web servers.

A trust model for wireless Internet must be highly scalable, fault-tolerant, and cost-effective in trust-path discovery and in mapping the security policy. At USC, a new WPKI architecture was proposed using a bridge CA cluster to achieve the security goals. Another advance lies in *distributed intrusion detection system* (DIDS) for protecting exposed Intranets or clusters of computers from malicious attacks. We developed the DIDS with dynamic policy update against changing threat patterns or varying network conditions.

Distributed security can effectively counteract both external intruders and insider attacks. XML, IDS, mobile agents, RMI, and CORBA are assessed as policy-update mechanisms to achieve dynamic security. The optimal choice of the mechanism depends on the tradeoffs among operating *speed*, Intranet *scalability*, host *robustness*, and the *security* level demanded. The WPKI and DIDS technologies benefit not only *M-Commerce* (mobile E-Commerce), but also pervasive computing applications in general.

1. Introduction

Security, scalability, reliability, and accessibility are the basic requirements in using the Internet and intranets for E-commerce or pervasive computing [11, 30]. This presentation covers recent advances in these frontier areas. Figure 1 illustrates the scalability versus security levels in protecting multicomputer clusters, LAN-based intranets, and metacomputing grids.

In a 2-dimensional cyberspace, today's clusters are poorly protected with scalability limited by security holes. Most intranets or LANs are protected only by boundary firewalls with a static policy. We demand a much tighter security with sophisticated IDS and PKI systems that can scale from clusters to LAN-based intranets and metacomputing grids. [10, 28].

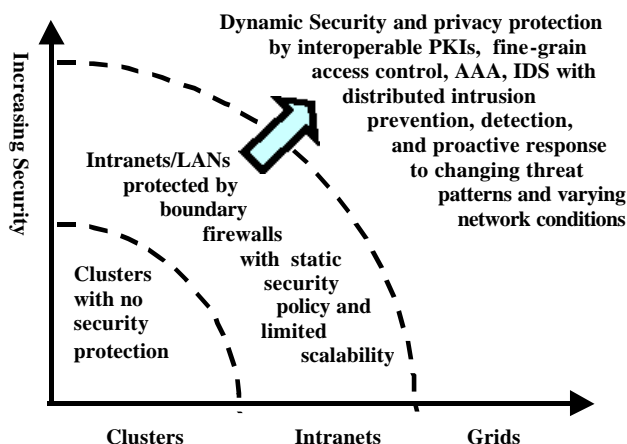


Figure 1. Securing clusters, intranets, and grids with PKIs and distributed IDS to achieve dynamic intrusion response with frequent security policy reconfiguration

To protect clusters, intranets, and computing grids, we must reconfigure the security policy dynamically to provide proactive intrusion detection and response. These security projects focus on *M-commerce* (mobile E-commerce) and pervasive computing applications.

The materials presented here are based on research findings at USC Internet Security and Pervasive Computing Laboratory during the last 3 years. For original results, the readers are referred to several of our earlier papers and presentations [7, 11, 12, 13]. The talk also brings up several new ideas from ongoing research projects on network and cluster security at USC.

Wireless Internet Environment: With the advent of 3G cellular RAN (*radio-access network*) and wideband WLAN (*wireless local-area network*) technologies, mobile Internet access demands the integration of all-IP-based voice/data communication with wireline, wireless, infrared, and satellite technologies with seamless interoperability [28].

Figure 2 shows the internetworking environment for 3G all-IP-based voice/data communication. Both wired

and wireless technologies are applied to access the Internet and intranet resources. For M-commerce and pervasive computing, one should focus on both ends of the mobile wireless world. Core competence must be established at both ends. Based on our prior experience in Linux clustering at USC, we are building a wireless gateway at the mobile Internet edge to support WTCP, WPKI, and AAA (*authentication, authorization, and accounting*) operations.

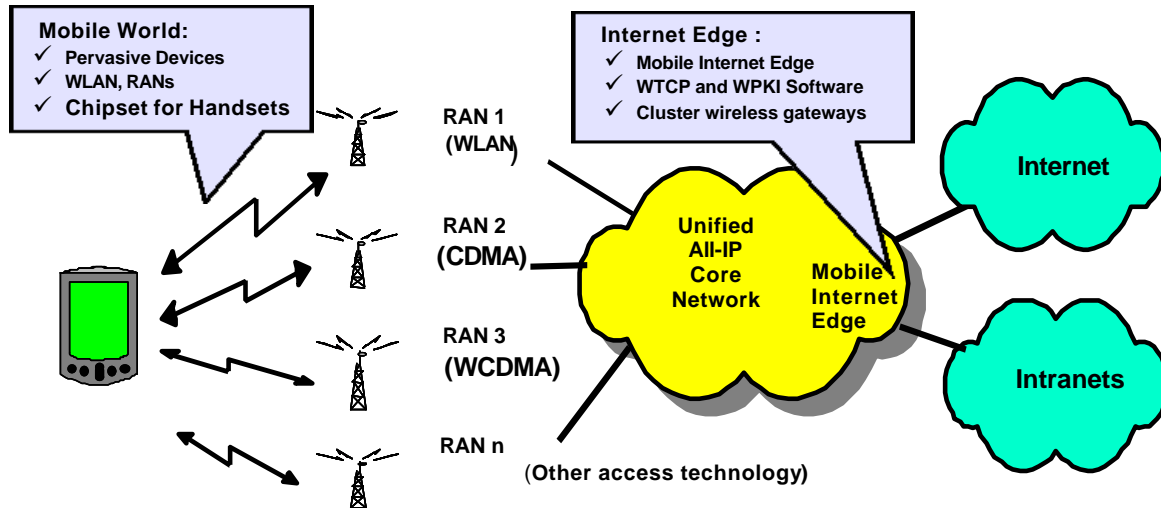


Figure 2 Wireless access of Internet in M-commerce and pervasive applications
(RAN: Radio-access network, WTCP: wireless TCP, WPKI: wireless PKI)

2. Wireless Public Key Infrastructure

Today's wired and wireless PKI systems are far from being interoperable [24]. This has hindered a satisfactory *wireless public key infrastructure* (WPKI) [6] to emerge for transaction security over the Internet. A trust-based approach was recently adopted at USC [13] to solve the wireless Internet security problem.

This work benefits mobile device users, wireless network operators, and Internet service providers in all-IP voice/data communications. In 1997, the *Wireless Application Protocol* (WAP) Forum was formed to upgrade the Internet standards for the wireless community.

The WAP specifies Internet access through mobile phones and other wireless devices like PDA. The transport layer protocol for security control in WAP is the *Wireless Transport Layer Security* (WTLS) [16], an optimized version of the TLS for narrow-band wireless networks.

Figure 3 illustrates the mobile computing environment using the WPKI architecture. The pervasive computing environments include the use of all sorts of mobile devices such as smart phones, PDAs, and handheld notebooks [8] over the 3G radio-access networks, WAP networks, or the IEEE 802.11 WLANs [28].

The WPKI arms the mobile user, the wireless gateway such as the PDSN used in CDMA 2000 and the GGSN in the UMTS networks, and the service provider with digital certificates. Whenever a mobile device requests a connection, it is secured by the WTLS protocol at the client side.

The gateway translates the request with encodes and decodes. The message is sent to the web server with SSL protection over they wireline Internet. It is essential to preserve the confidentiality, data integrity, and authenticity of data to prevent fraud, abuses, or malicious attacks from intruders.

WPKI Design Objectives: Conventional PKI works on wireline-connected Internet. Wireless PKI provides a secure and trusted trading environment. In both cases, the following four security requirements must be met using cryptography, digital signatures and certificates [6].

- **Confidentiality of exchanges** – Make sure that nobody can listen in.
- **Authentication** – Certify the identities of both clients and servers.
- **Data Integrity** - Assure that information is not tampered on its journey.

- **Non-repudiation of transactions** - Assure that the transaction is legally binding.

In the past, PKI has been heavily applied in E-Commerce applications over the wirelined Internet [14, 19]. Overviews of trust models for building PKI have been given by Linn [17], Manchala [19], and Perlman [21]. Kagal, et al [16] has studied trust-based security issues for pervasive computing. The wireless PKI issues and WTLS

were studied in [6,13, 21]. The *bridge certification authority* (BCA) has been treated in [1, 25].

Recent studies of PKI interoperability were reported in [9, 13]. One can find the Certicom PKI portal design in [4]. A recent NSS Group Test of various commercial PKI systems can be found in [21]. Recent R/D and performance evaluation reports of IDS can be found in [5, 18, 20, 21, 26].

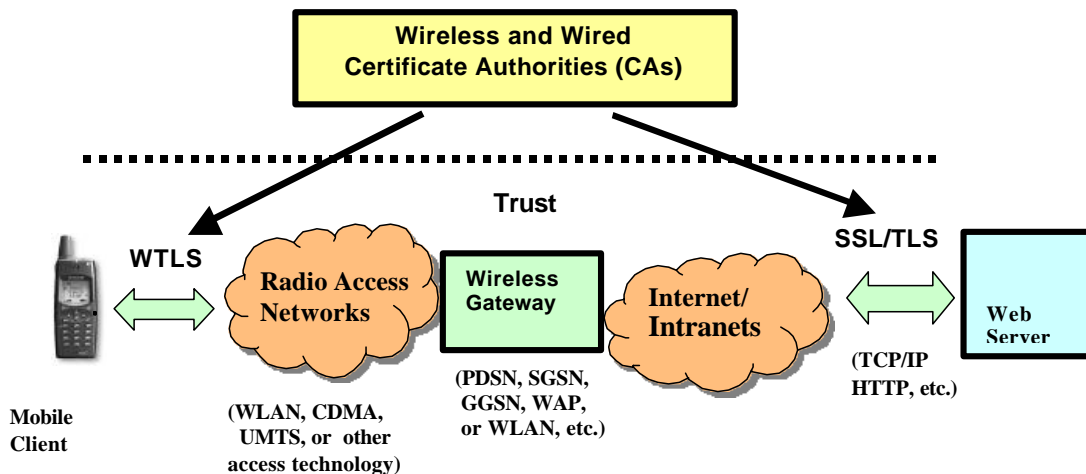


Figure 3 Pervasive computing with wireless Internet access protected by PKI using WTLS protocol at client side and SSL/TLS protocol at server side

PKI Growth Model: The global security market will grow to \$14 billion by 2005, 15% of which from the traditional wireline PKI and 10% from using the WPKI. Both wired and wireless PKI are needed to provide a secure and trusted transaction environment. Unfortunately, wired and wireless PKI systems are not being designed to be fully interoperable to each other.

Future mobile computing systems will demand higher quality of services than that available today. According to a 2001 forecast by Data Monitor, the PKI-based security market for mobile subscribers will experience a major increase in market share and growth in the next 6 years.

The forecast identifies the market growth in four areas: *PKI-enabled applications*, *wireless PKI*, *PKI with access technology*, and the *pure PKI*. The wireless PKI shows the sharpest growth rate in the next few years. This has motivated us to bridge the interoperability gaps between wireline and wireless Internet.3.

3. Wireless PKI Interoperability

In this section, we introduce the basic WPKI architecture and identify the differences between wired PKI

and wireless PKI platforms. Then we discuss the security demands in a mobile pervasive computing environment.

Basic WPKI Architecture: Traditional wired PKI components must be modified to work in the wireless environment with satisfactory performance. A new PKI portal serves as a wireless RA (*registration authority*). The CA (*certification authority*) must be modified for wireless certificate management. The new Bridge CA is needed for trust management across the boundary between wired and wireless PKI domains.

The wireless gateway serves as a bridge between the WTLS and SSL security protocols from the mobile client to the wireless services provider (e.g. web server). It takes SSL-encrypted messages from the web and translates them for smooth transmission over a wireless network using the WTLS security protocol.

Similarly, messages from the mobile devices to the web server are likewise converted from WTLS to SSL. The directory server serves as the certificates repository. In contrast to using web browsers on the Internet, mobile devices do not store their own certificates.

Instead, they store an URL for each certificate they use. The verifiers embedded in the URL are used to retrieve the certificate. The deployment of the directory is

determined by the application scale, customer budget, and security requirements in the WPKI.

WTLS Protocol and Certificates: The WTLS and TLS use different certificates to authenticate between the client and server. The WTLS operates over a datagram protocol and provides an end-to-end security. The WTLS must be designed to minimize the protocol overhead and apply data compression beyond those used in traditional SSL/TLS.

WTLS mini-certificates are similar to the X.509 but much smaller and simpler for use in resource-constrained mobile devices. The WTLS is optimized for use over low bandwidth communication channels. For wireless Internet access, WTLS provides confidentiality, authentication and data integrity, while non-repudiation is provided by digital signature. Each WTLS certificate holder will generate a pair of keys, a *private key* and a *public key*. These uniquely issued key pairs are linked mathematically using asymmetric cryptography.

Security in Mobile Computing: Mobile computing includes pervasive computing and M-Commerce. Both are hot application areas facing serious security challenges. These security problems spread over transactions, infrastructure, applications, and database systems. The WPKI development is meant to attack the wireless network security problem from all dimensions in an integrated manner.

For an example, a mobile client sets up a WAP site using a *Wireless Mark-up Language (WML)* page. The client must specify the tags to call a signing function. The client downloads a page, fills in a form, and clicks on a button associated with the signing function. To enable the signing function, the user enters a pin code to unlock his private key, stored in the *Wireless Identity Module (WIM)*.

When the M-commerce vendor sees the signed data, it needs to verify the authenticity of the user and to check the integrity of the data. All of the above transaction processes are done via the digital certificate from a repository owned by the CA.

The integrated solution enables consumers to sign high-valued transactions with a simple mobile device, like a PDA. The WPKI enables the retailing world by non-repudiation, permitting legally binding transactions to be performed in mobile pervasive environments.

WPKI cannot function properly if it is not interoperable between wirelined and wireless networks jointly used to access the Internet. Both demand secure downloading CA roots and server and client certificates. At the highest level, interoperability covers technology, business and sometimes even legal issues.

According to PKI Forum [24], the interoperability framework can be conducted at three levels: the *intra-domain level*, *application level*, and *inter-domain level*.

(Fig.4). These three levels have increasing level of sophistication in their interoperability testing requirements.

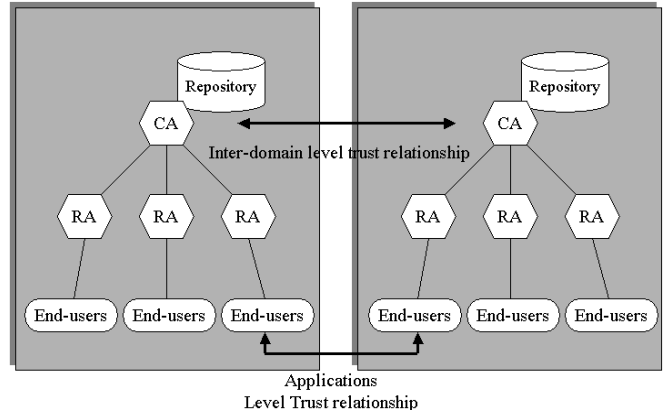


Figure 4 PKI Interoperability at the inter-domain and application levels

Intra-domain interoperability refers to component-level relationships, such as peer-to-peer or superior and subordinate trust relationships. This level requires testing security components that fall under the control of a common administrative authority.

Application-level interoperability is checked between two peers, regardless of the supplier of the application or any ancillary infrastructure components used. For interplay between scalability and interoperability, we assess below the interoperability at the Inter-domain level.

Inter-Domain Interoperability: This deals with the issues and options between two isolated PKI domains. A PKI domain is an autonomous infrastructure deployed within an organization or enterprise. Therefore, inter-domain interoperability essentially tests between two PKI domains.

This is the most complex level, involving the cooperation of multiple PKI domains. Inter-domain interoperability involves a number of challenges; all are technology and policy related. One must figure out a method for establishing trusts relationships between the PKI domains.

The PKI-related information in one domain must be made available to another and vice versa. Each PKI domain must agree to adhere to certain policies and to have mechanisms in place to enforce adherence to the agreed-upon policies.

With the physical and operational differences between wireline and wireless PKI domains, their inter-domain interoperability posts a major R/D issue. This will affect both server-side and client-side PKI designs. Without significant advances in this interoperability level, the technology of trust cannot move seamlessly into the wireless era.

Listed below are 10 most important technical issues identified in [13] for establishing the inter-domain PKI interoperability. Today's PKI systems implemented only a small subset of these requirements at the Intra-domain or application levels. Inter-domain interoperability is a superset, demanding smooth operations at the lower levels as well.

- (1) Common protocols, message and certificate format between different PKI domains.
- (2) Common algorithms for entity authentication and data protection between PKI domains
- (3) Data encapsulation and encoding formats must be compatible between PKI domains
- (4) Support the storage and retrieval of certificates between the repository and PKI domains
- (5) Private keys must be accessible by authorized end entities securely regardless of storage method (e.g., software, smart card, or hardware token)
- (6) Certificate information must be compatible and consistent not to hinder interoperability
- (7) Support multiple applications from different vendors across the PKI domains
- (8) New methods are sought for establishing the trust relationship between PKI domains
- (9) Enhanced transparency in PKI-related information sharing among multiple domains
- (10) Adherence to the agreed-upon security policies among cooperating PKI domains

4. Trust Models for Mobile WPKI

This section assesses the appropriate trust propagation models that can be used to enhance security in mobile E-transactions and pervasive computing. We assess the strength and weakness of five trust propagation models for mobile wireless PKI operations.

Trust-Propagation Models: Alternative PKI models are evaluated below for trust propagation. In these models, cross-certification of one CA by another is the basic technique to bridge truest relationship between PKI domains. Some models contain subsets of CAs to cross-certify one another. We assess the following PKI models in Table 1 for cross-certification operations.

- **Hierarchical PKI:** One or more trusted root CAs issuing certificates top down.
- **Mesh PKI:** Built with of cross-certified CAs with one-level of subordinate entities
- **Hybrid PKI:** Using bilateral cross-certification between hierarchical and mesh PKIs
- **PKI built with Bridge CA:** Interconnect PKI islands by a central authority of cross certification.
- **PKI built with Trust Lists:** Providing client systems with a set of trust lists, enabling remote entities to discover the trust path driven by verifiers.

Table 1. Assessment of Five Trust Propagation Models for Building PKI Systems

Trust Models	Service Growth Scalability	Trust Path Construction	Interoperability among PKIs	Directory Dependence
Hierarchical PKI	Medium with a top-down growth	Simple down from the root	Weak beyond the root CA	Low due to one or fewer roots
Mesh PKI	Low with a pairwise growth	Hard due to multiple routes and iterations	Good with limited number of PKIs	Robust but highly dependent
Hybrid PKI	Medium with a top-down growth	High with simple path under multiple routes	Good with limited PKI domains	Medium with limited PKIs
PKI using Bridge CA	Very high with Bridge CAs	Medium with all paths traversing the bridge	Very good with many PKI domains	Very robust with a cluster architecture
PKI using Trust Lists	Medium, recognized by verifiers	Simple but limited within local trust lists	Fair due to extensive management	Robust and low dependence

Our assessment is centered on the issues of service scalability, fault tolerance, and the implementation complexities. These attributes determine the performance and efficiency in trust path construction, certificate validation, and security policy management.

Among the five trust models, we prefer to use the bridge CA for its virtues in cost and performance. This trust model has the strongest scalability and interoperability, because many different PKI domains can be bridged together through a central authority of cross certification.

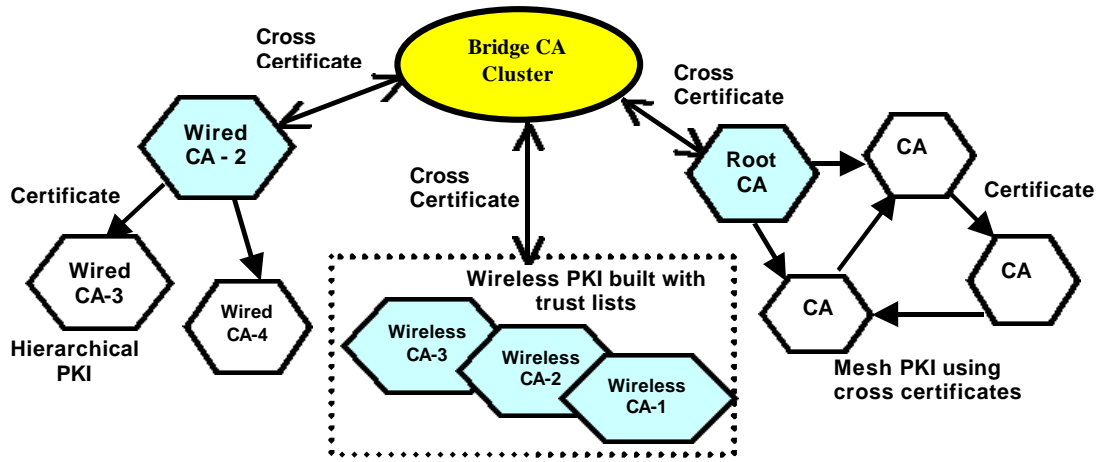
The Bridge CA Architecture: Bridging is the most promising model between wired and wireless PKI domains. This model outperforms other trust models in scalability, interoperability, and robustness. The cluster is a powerful architecture to implement the bridge CA model for trust propagation across multiple PKI domains.

The bridge CA model scales well with the demand of services over wired and wireless PKI domains. Each participating PKI domain sets up a cross-certificate with the central Bridge CA. The trust relationship between BCA and participating CA is peer to peer.

To interconnect n PKIs, $2n$ cross-certificates are needed. This model interconnects PKIs with heterogeneous

structures, protocols, and devices. An example trust-path is shown in Fig.5, which is constructed from CA-3 in the wired hierarchical domain to the wireless CA-1 in the wireless domain built with trust lists. The participating domains can assume different PKI structures such as the hierarchical, trust lists, and mesh.

The *trust anchor* of the bridge CA locates itself. The inter-domain trust path must traverse through the bridge CA with cross-certificates carrying the policy mapping, path constraints, etc. By setting up the central trust bridge, the length of trust path is effectively shortened, thus the overhead can be greatly reduced. The robustness of the bridge CA is powered by cluster architecture.



Example Trust Path construction from wired CA-3 to wireless CA2:

Wired CA-3 → Wired CA-2, Wired CA-2 → Bridge CA,
 Bridge CA → Wireless CA-1, Wireless CA-1 → Wireless CA-2

Figure 5 Trust propagation by bridging various PKI domains using bridge CA cluster over wired and wireless networks

Policy Reconfiguration: In most cases, wireline policies differ from wireless PKI policies user registration, acceptance of hardware/software devices, and certificate storage, etc. These differences will be enlarged when the number of participating PKI domains increases.

Through BCA, multiple policy mappings are represented in a single cross-certificate. Each principal CA issues a cross-certificate to the BCA and asserts its own domain policy to map the BCA policies. The efficiency of cross certificate relies heavily on the interoperability between different PKI domains.

This mapping does not care which case the policy is applied to, wired or wireless, because it can be configured to work with both. The bridge CA architecture can scale and provide inter-enterprise support to a large scale. This enables interoperability to enforce dynamic security [3].

5. WPKI Extensions from Wired PKI

We treat the WPKI as an expansion from existing PKI system to build up the trust relationship in a wireless environment. In order to fully leverage on existing PKI systems and Internet resources, the design must be scalable. We adopted a cluster architecture, which has been built at the Internet and Cluster Computing Lab at USC.

Wireless PKI Extensions: In [13], we have examined the extensions of wireline PKI to a wireless environment, as repeated in Figure 6. The thick dash line separates the elements built in conventional wired PKI from those to be added in the wireless expansion.

The WPKI provides a basis for establishing a key to encrypt a client-server session. The server side authentication is crucial in many application contexts. Without server side authentication, there is no assurance

that the client is sending confidential information to the appropriate party.

The WPKI system developer must deliver security features, multi-OS platform, and scalability support. The WPKI platform can be designed to run with popular operating systems such as the Windows, Solaris, and Linux.

Nice features in different operating systems should be explored to meet the special requirements in mobile applications. The scalability of the proposed WPKI system is enabled by the clustering and middleware technologies.

New Components in WPKI: Three functional units are considered new in the WPKI expansion: namely the wireless PKI portal, wireless CA, and bridge CA in Fig.6. These units must be specially designed for WPKI. The PKI portal is essentially a wireless RA. The wireless CA is mainly used for certificate management in the wireless environment.

Current mobile devices have limited resources to validate the certificates. A server-side *online certificate*

status protocol (OCSP) has been suggested, but fully implemented yet. The wireless CA must manipulate both WTLS and X.509 certificates. It must be specially designed for robustness and interaction with all other CAs.

Wireless Security Services: The WPKI designer must specify the encryption algorithms, server and client certificates, and key WAP and WPKI features desired. Openness is a primary requirement to achieve the interoperability.

The WPKI and WTLS support wide classes of encryption, hashing, and key exchange algorithms. Strong authentication demands 2048-bit RSA encryption, 256-bit ECC and 128-bit symmetric encryption algorithms.

The USC team applies enhanced elliptical curve and AES algorithms [13]. This will expand the trust in securing communication between wireless and wired networks. The class 2 services support only client-side authentication, while the class 3 services support two-way authentication at both server and client sides. Class-3 services are more desired in a wireless environment.

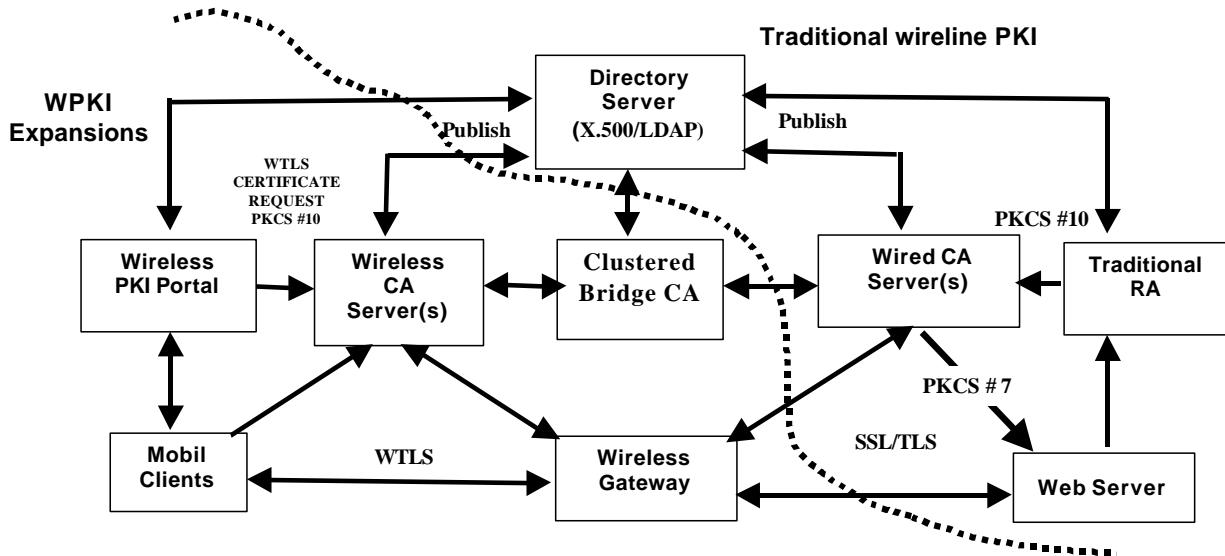


Figure 6 Wireless PKI expansions from the traditional wireline PKI [13]

6. Bridge CA and Trust Middleware

We consider unique features specially designed for the bridge CA cluster. In Figure 5, we have shown lightweight bridge CA cluster architecture to serve as the core interoperability component between wired and wireless CA's. Through the establishment of cross certificate pairs, we can build up the peer-to-peer trust relationships between the principal CA's in both wireless and wired domains. In particular, we address the issues of using clustering technology to implement the trust-management middleware.

Cluster Architecture for the Bridge CA: The bridge CA must be able to support a large number of mobile users. The trust path construction procedure must be simple by which all non-local paths traverse through the bridge with minimum cost. The directory dependence should be reduced, even the directory database becomes very large.

The pairwise growth pattern in the bridge CA model makes it highly scalable in modular growth. We use the *Bridge CA* cluster for managing the trust relationships across the wired and wireless boundary. Traditionally, the

bridge CA establishes peer-to-peer trust relationships with the end user communities using the Federal Bridge CA [1].

The FBCA lacks the compatibility with wireless devices and protocols. Different from FBCA, the clustered bridge CA will enable the authentication between wired Internet user community and mobile user community. It links both wired and wireless PKIs at a single hub or bridge cluster.

In Table 2, we summarize the important security features and compliance of standards in the design and operation of the bridge CA, the PKI portal, and wireless CA. For interoperability among all functional units in the WPKI system, we suggest to use the same set of cryptographic algorithms. The vision refers to the perspectives from participating PKIs, clients, or other PKIs, which are not directly connected to the bridge CA cluster.

Table 2 Design Choices of Bridge CA, Wireless CA and Wireless PKI Portal [13]

Functional Subsystems	Vision	Security Features and Actions	Standards Compliance
Clustered Bridge CA	Participating PKI's	Cross-certificate issue, update, renew, revoke, path discovering	X.509 V3 certificate, WPKI certificate, X.500 LDAP
		Security policy manipulation	SDN.801-based SPIF
		Inter-domain access control	Directory, Attribute Certificates
Wireless CA	Client	Certificate Issue, update, renew, and revoke	X.509 V3 certificate, WPKI certificate, X.500 LDAP
		Client certificate request handling	PKCS #10
		Certificate status inquiry/response	OCSP, RFC2560
	Other PKI	Interact with other X.509 PKI	CMP, CMC
Directory certificate publish		X.500 LDAP	
Wireless PKI Portal	Client	Certificate request forwarding	PKCS#10
		Wireless certificate requests	WMLScript
		Client certificate URLs in LDAP	X.500 LDAP
	Other PKI	Interact with X.509 PKI	CMP, CMC
Encryption and Decryption Algorithms recommended		Public Key Cryptosystems	RSA, ECC
		Symmetric Encryption	AES, Triple DES, RC4, RC5
		Hashing Algorithms	MD2, MD5, SHA-1

Functionalities and Advantages: Functionalities of a bridge CA cluster are identified below first. Then we identify the advantages of the clustered bridge CA architecture:

- Support the cross-certificate registration from the wireless CA's
- Support the cross-certificate registration from wired CA's
- Support WTLS certificate and X.509 certificate path discovery and trust management
- Cross-certificate/CRL management
- Interoperability with directory through LDAP
- Policy reconfiguration and management

The bridge CA cluster in our combined PKI and WPKI system has the following four distinct advantages over the Wireline Bridge CA architecture:

- **Scalability:** The Bridge CA cluster can scale freely in establishing the trust relationships between multiple CA's associated with different PKI domains.

- **Heterogeneous certificates:** WTLS mini Certificate and the X.509 Certificate are supported and thus offering higher application potential.
- **Higher availability:** The BCA cluster architecture has failover and recovering capability after any failure of any individual root CA attached to the bridge.
- **Low implementation cost:** The certificate path discovery and interoperability support grow in a pairwise peer-to-peer fashion, thus reducing the implementing costs.

Middleware for Trust Management: Based on the clustered BCA design, a trust management middleware package must be developed to achieve the desired scalability and unified interfaces. The middleware glues all sorts of *processing engines* to run on the Linux cluster nodes at USC.

Four processing engines (servers) are shown at the top boxes in Fig. 7. The functions of these engines include the management of cross certificates, CRL, directory or

LDAP, and policy configuration. Both the system administrator and users will benefit from this clustered bridge CA design, because the cluster is highly scalable and available.

The user registration to the bridge CA system is simplified. The process to export, import, or update the policy is more efficiently carried out. The administrator can add or remove redundant cluster nodes to yield higher performance without affecting the front-end users

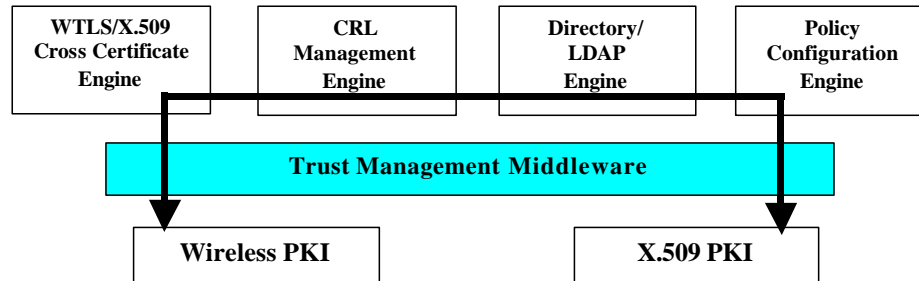


Figure 7 Middleware for trust management on the Bridge CA cluster

Scalable WPKI Services: With above designs of the bridge CA, the PKI portal, and the wireless CA, we summarize scalable WPKI services below. To achieve these, we need to use a *reference* and *replication scheme* to upgrade the directory system.

The purpose is to deal with the ever-increasing scale of the wireless certificate databases. Itemized below are security services enabled by the scalable WPKI architecture presented in previous sections.

- (1) Performing the user identification through direct confrontation
- (2) Providing user with registered ID and password
- (3) Providing server-side authentication
- (4) Provides client-side authentication
- (5) Sign certificate request, verify digital signature, and send the keys to PKI portal
- (6) PKI portal confirms the ownership of the digital signature
- (7) Wireless user gets the certificate and exchanges with digital signatures

7. Dynamic Security for Intranets

In the past, most Internets are protected by enclosed networks, which are isolated for private use only. A single gateway firewall is often used at the front-end to repel malicious attacks from external sources. The baseline assumption in the gateway is to trust all cluster nodes and distrust all external hosts.

With the sophistication of today's hackers, an external attack can easily penetrate the gateway and become a threat from inside. All Intranet hosts may compromise quickly with a domino effect, if the attack is initiated from an internal node of the Internet.

At USC, we adopted a *distributed micro-firewall* approach [2, 7, 12] to solving the security problem threatening all network nodes. In a cluster of computers forming the Intranet, cluster nodes appear mostly as PCs, workstations, or servers. The gateway firewall protects a cluster from external attacks, but not from insider attacks.

Dynamic security is a strategy to provide security awareness and adaptability to address runtime policy changes. Integrated special security toolkits are needed with some intrusion detection frameworks to automate the host-level responses to intrusions. Database and server reconfiguration methods were suggested in for implementing adaptive security policies.

We present below the new approach developed at USC for implementing dynamic security with micro-firewalls. We suggest using mobile agents [7, 15], XML [28], and RMI (*remote method invocation*) [7, 12] for policy reconfigurations. We present below the key concepts of micro-firewalls and distributed IDS design at USC.

We build functional mechanisms built in the micro-firewalls in Linux kernel. We use mobile agents for auditing records, detecting anomalies, and reporting intrusions. For security policy update, we suggest the use of XML or RMI package to report and broadcast.

Distributed Firewalls: Bellovin [2] pioneered the concept of *distributed firewalls* to alleviate some of the above problems. Distributed firewalls impose access control at individual hosts rather than at the gateway of the network.

Imposing access control at individual hosts leads to fine-grained security processing with no restrictions on the network topology. They implemented their prototype system, used the IPsec policy language and the KeyNote trust management system.

The end-to-end encryption was used on all hosts. However, establishing the IPsec connections between all

hosts pays a high runtime overhead. Our micro-firewall approach has much lower overhead and lower cost to implement. This enables real-time intrusion detection and response in a cluster or Intranet environments.

A highly secured cluster domain must be supported with scalable infrastructure and recoverability from intrusions or system faults. In addition, the framework must be able to monitor the software agent's behavior to assure trusty and reliable operations.

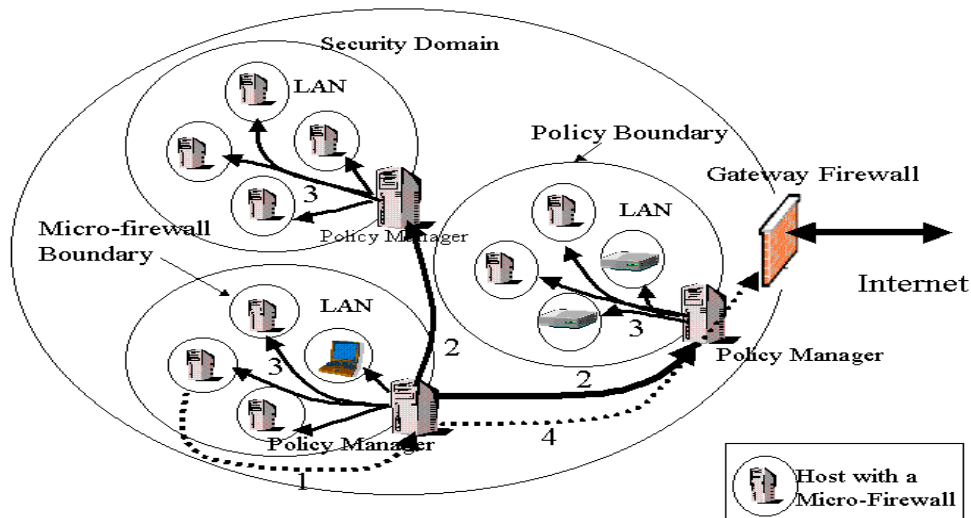
Distributed firewall compensates what cannot be done by the conventional gateway firewalls. In a cluster environment, both are used jointly to provide double protection and aggressive response to intrusions from all possible sources, either inside or outside.

Collectively, they enable dynamic security and adaptive intrusion detection and responses. They are also

more robust, fault-tolerant, scalable, and cost-effective to protect exposed Intranets or clusters.

Dynamic Security: Security policies enforced in existing Intranets are mostly static. The static policy cannot cope with the dynamic changes in threat patterns [3, 22, 26]. Policy changes in these Intranets cannot be done in real time and often involve human experts in the loop.

Dynamic security demands on-line policy change, when new threats or intrusions are detected. Adaptive intrusion responses are expected in such an Intranet. This cannot be achieved by using the single gateway firewall alone. Figure 8 shows how to use distributed micro-firewalls and host-based IDS in steps to achieve dynamic security over an Intranet containing multiple security domains. The 4 steps are carried out sequentially as specified below.



- Step 1. Intrusion detected by micro-firewalls and host IDS**
- Step 2. Intrusion response through policy updates on all policy managers**
- Step 3. Policy update on all micro-firewalls in each policy domain**
- Step 4. Change screening rules in the gateway firewall**

Figure 8. Dynamic security policy updated by host-based micro firewalls and IDSs under the coordination of the policy managers and gateway firewall

The administration of the central security policy is distributed to all cluster nodes under the coordination of the *policy manager*. A *gateway firewall* is installed at the front-end, playing the role of screening between the cluster network and the external world of the Internet. Since individual nodes act as enforcers of the central security policy. All the cluster nodes form a single security domain. A *micro-firewall* is built on each node by OS kernel extensions aided by some intrusion detection and policy updated mechanisms.

The distribution of security functions removes some of the constraints associated with conventional gateway firewall. This distributed architecture is meant to

cope with insider attacks. In addition, the cluster is supported by a policy update mechanism, that responds to new attacks dynamically.

The functions of intrusion prevention, detection, and responses are now distributed to three levels of security control, namely the *gateway firewall*, the *policy manager*, and *micro-firewalls*. Collectively, they carry out the security enforcement and update the security policy dynamically.

The main purpose is to repel intrusions from all sources in real-time. In particular, we protect the Intranet resources including the network hosts, file systems, software processes, access control, and system administration, etc

8. Distributed Intrusion Detection System

We consider Intranets containing a large number of computer nodes, which are prone to frequent intrusions from all possible sources. We specify below the functional characteristics and assess the structural complexities of security control in both levels.

The gateway firewall plays essentially the role of *intrusion prevention*. All Intranet nodes work with the policy manager to implement a *distributed intrusion detection system* (DIDS).

In Fig.8, we denote the micro-firewalls as level 3, the policy manager as level 2, and the gateway firewall as level 1. We treat the attacks from external sources (Internet) differently from insider attacks. At level i , we define two sets of screening rules, S_i and T_i , applied for external attacks and insider attacks, respectively.

These rule sets follow the set inclusion properties: $S_3 \subset S_2 \subset S_1$ and $T_3 \supset T_2 \supset T_1$. The numbers of interacting hosts involved roughly estimates the set complexities at different levels.

In gateway security, we see a static security policy applied. This induces a performance bottleneck on the gateway, since the rules are set at the maximum screening level. This may hinder normal network-based applications.

The drawbacks of a static security policy can be overcome by an adaptive approach as illustrated below. In an adaptive scheme, the gateway firewall is not configured to have the maximum-security protection.

The cycle of threat detection, software vulnerability, safeguard choice, and adaptive response form a feedback loop in Fig. 9. The primary functions of the micro-firewalls are to monitor local events and to audit records relevant to penetrations. The policy manager is responsible for maintaining the IDS and initiate policy changes.

The gateway changes the safeguard by taking new policy issued from the IDS. The security safeguards can be chosen from rule changes in distributed firewalls, or by adding authentication, or by access control, or by some encryption countermeasures.

The manager located in the DMZ implements the adaptive security policy. The manager receives reports from the cluster nodes and applies some rules to choose new safeguard and elevate the level of screening at micro-firewalls as well as in the gateway level.

To achieve this kind of adaptive security policy, the system should have mechanisms at the nodes and the manager level to change security policies and to enforce

them. The IDS functions must be distributed to all cluster nodes under the supervision of the manager.

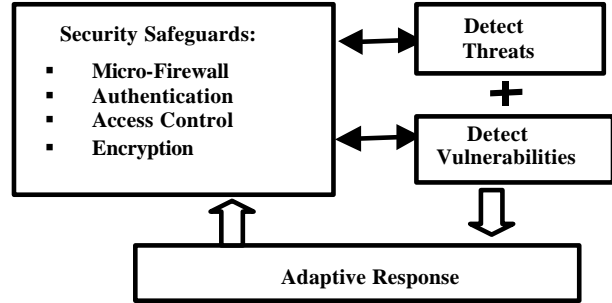


Figure 9. Dynamic security update cycle: Detecting threats and vulnerabilities, learning from intrusion patterns, changing security safeguard, and active intrusion response to all attacks.

The gateway firewall does not involve in policy change, rather it serves as a responsible guard to screen the traffic to enter the Intranet domain. Figure 10 shows that the micro-firewall as a functional module at the network-access part of the OS kernel. The module is placed between the TCP/IP stack and system call interfaces.

The TCP/IP stack is directly connected to network cards. The system calls interface with application programs. The micro-firewall module consists of three functional blocks: the *packet filter*, *anomaly detector*, and *access logging* as specified below separately.

We are concerned about detection-specific audit records. For example, we want to save the critical processes timely or to stop the intruder from getting the root access privilege. In addition to using the Snort for IDS, we explore the Psionic PortSentry tools for dynamic intrusion detection. This is a port scanner that takes an active stance to shut down attacking hosts while provides a reconfiguration.

Attacking hosts are denied access to the cluster by dropping local routes or adding the host to a TCP deny file. In addition, we apply the LogSentry to accumulate log generated and the Tripwire for file integrity checking. For the micro-firewall, we use the IPtable in Linux hosts.

9. Security Policy Update Mechanisms

We consider alternative mechanisms for updating security policies with micro-firewalls. These mechanisms are assessed below for the purpose of dynamic policy update. First, let us specify the goals of dynamic policy update. Then we compare mobile agents, CORBA, and RMI middleware for policy reconfigurations at individual host, policy manager, and gateway levels.

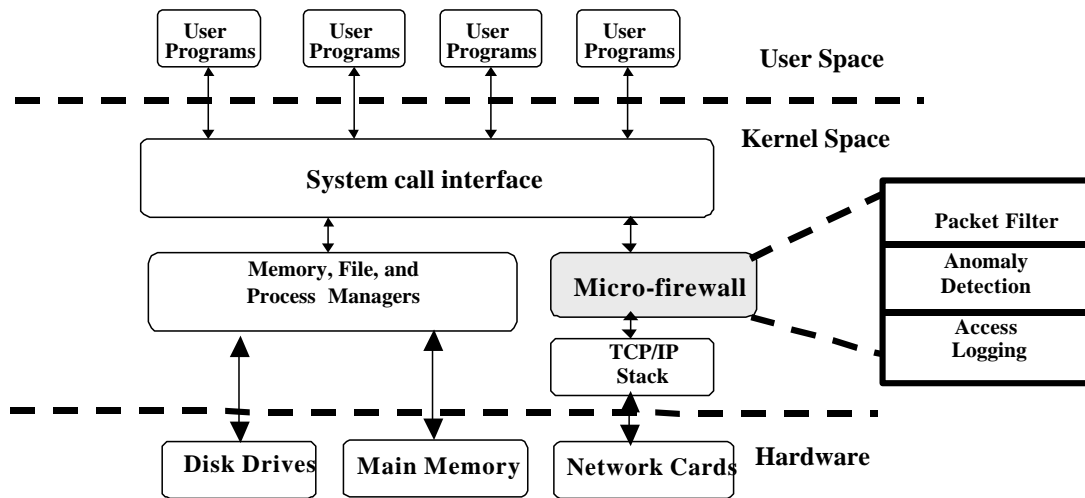


Figure 10. The implementation of a micro-firewall at the kernel space of a Linux host

Policy Update Goals: Dynamic security demands frequent policy changes. The main items to be changed include the rule sets in micro-firewalls, managers, and gateway. They differ in granularity and scopes.

One primary goal is to maintain the critical mission functionality. At the host level, the security update must reduce the risk of further penetration or stop the widespread of the attacks. These timely actions must not take at the expense of terminating critical processes.

At the security manager level, major goals in policy reconfiguration include the reloading of new security database and expanding the state and mode of operations for the manager. Inter-manage communications are needed to exchange policy update information.

The managers achieve distributed intrusion detection collectively and cooperatively. Further, they prevent the spread of threats beyond each security domain. All managers also required interacting with the gateway firewall to make changes in the global security policy accordingly.

Comparison of Three Mechanisms: Table 3 compares these two mechanisms for security policy update. They are compared along four functional features: namely the ability of central policy coordination, response time to policy change, security of the mechanisms themselves, and process termination. The table entries demonstrate the fact that agents are autonomous, require no coordination after dispatched, and always terminate in the policy-update process. These are positive features to implement the policy update process.

However, agents rely on authentication and encryption to protect its own security. For this reason, agents are easier to be attacked by other agents or hosts.

Mobile-agent systems save network latency and bandwidth at the expense of imposing higher workload on their hosts.

Agents are often written in a slow interpreted language for portability and security reasons. The agents must be injected into an appropriate execution environment upon arrival. Mobile agents may take longer time to accomplish a task, since the timesavings from avoiding intermediate network traffic is less than the penalty from slower execution.

Fortunately, significant progress has been made on just-in-time compilation (most notably for Java) and software fault isolation. These allow mobile codes to execute as fast as natively compiled codes. Nearly all mobile-agent systems allow a program to move freely among heterogeneous hosts.

The agent code is compiled into some platform-independent representation such as Java byte codes, and then either compiled into native code upon its arrival at the target machine or executed inside an interpreter. For mobile agents to be widely used, the code must be portable across mobile-code systems. Making agent code portable across platforms requires a standardization effort.

CORBA needs coordination by the security managers and demands the ORB support. However, the CORBA is faster than agents requiring less execution time. CORBA middleware is more secure by using the CorbaSec.

An RPC-like semantics is needed to terminate the policy update process. The use of CORBA for policy update includes the provision of a multi-language and multi-platform environment, distributed object infrastructure, location transparency, and network transparency.

Table 3. Comparison of Agents, CORBA, and RMI for Security-Policy Update [7, 12]

Capabilities	Mobile Agents	CORBA Middleware	RMI Middleware
Central policy coordination	Agents are autonomous and require no coordination once dispatched	The policy manager in each domain coordinates all communications	Policy manager acts as the RMI registry coordinating all communications
Reaction time to policy change	The time increases with the number of agents dispatched.	Faster than agents or RMI to react to a policy change	RMI is slower than CORBA and is faster than agent based system for policy updates
Hosts fortified with micro-firewalls	Agents carry most mechanisms required to update security policy	Requires the ORB middleware support on all hosts in the Intranet	Requires JVM to be present on all the hosts.
Security Mechanisms	Use authentication and encryption. Still prone to attacks from other hosts.	Security is implemented with the CORBASec.	Security is the best among the three, implemented with the Java sandbox model.
Update Process Termination	Multiple agents used autonomously, Policy update always completed.	Implemented at application level using RPC-like semantics	Implemented at application level using RPC-like semantics

Relative Strengths and Weaknesses: The above policy update mechanisms are rated below in terms of *operating speed, scalability, security base, and robustness*.

(a) Mobile agents are strong in scalability and robustness. The number of active agents can be scaled up or down, depending on the demand and the variation of the network size. Robustness refers to the fact an agent could have multiple lives.

The agents can be easily created, suspended, terminated, or reborn dynamically. The major weakness of agents lies in its high overhead and lower security base themselves. Thus agents have the lowest speed and most vulnerable to be attacked by other agents or hosts.

(b) The major strength of CORBA is its high speed and lower overhead experienced. For applications that demand high speed in policy update, CORBA should be the choice. However, CORBA is rather weak in scalability and robustness. Among the three, CORBA is ranked in the middle in terms of base security.

(c) RMI has the highest base security among the three. This is due to fact that Java security is based on the sandbox model, which is much securer than either agents or CORBA. Like the CORBA, RMI is equally poor in scalability and robustness, when used in a large network environment. For speed performance, RMI is ranked in the middle.

We have also checked the use of Email, RPC, AC (*attribute certificates*) XML, SNMP, and HTTP for policy update. In particular, we find the use of AC and XML very attractive in key/certificate management and in the security-policy reconfiguration process.

10. Dynamic Response To DIDS

Figure 11 shows the architecture of an agent-based distributed intrusion *detection system* (DIDS) [7]. The building blocks of the DIDS are the policy managers (PM) in various security domains.

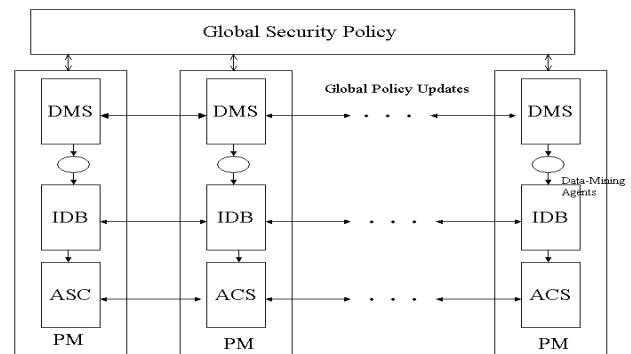


Figure 11 Architecture of a distributed intrusion detection system (PM: Policy Manager, DMS: Decision Making System, IDB: Intrusion Database, ACS: Agent Control and Security)

The ACS (*agent Control and Security*) consists of an *agent name system* (ANS) for establishing a unique name space of all agents. This ANS controls the access of the agents and eliminates malicious agents. The ACS needs to apply PKI to enable secure communication and agent authentication.

The IDB (*intrusion database*) keeps track the creation, entries, exits, and activity records of the agents, either from external or internal sources. The security

policies must be established depending on the levels of security required. The *decision-making subsystem* (DMS) determines the timing and countermeasures to be taken

This subsystem operates adaptively with respect to the changing environment. The *intrusion database* keeps track of the creation, entries, exits, and activity records of agents. The security policies are established depending on the levels of security required.

Consider a cluster fortified with micro-firewalls and a distributed IDS system in Fig.12. This example

illustrates the response to an attack in 3 steps: In Step 1, an external attack has penetrated through the firewall and hide in a cluster host. In Step 2, the intrusion is detected by the micro-firewall locally.

An RMI is dispatched to alert the IDS at the manager with the intrusion record. In Step 3, the manager updates the policy to cope with the intrusion. The larger is the cluster, the more overhead will be experienced in these active response operations.

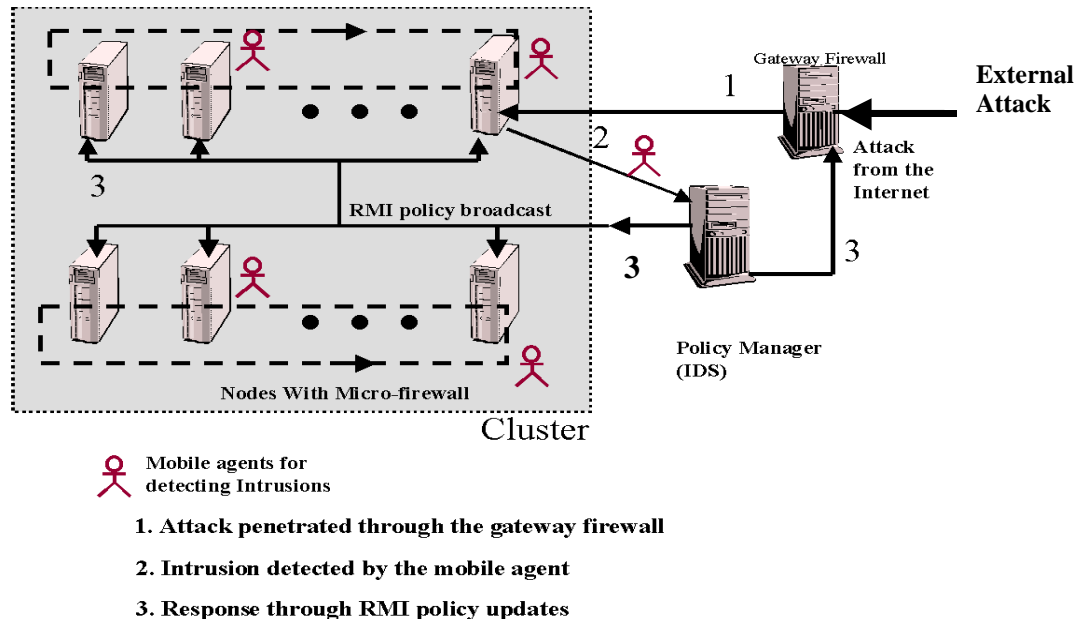


Figure 12 Distributed intrusion detection and proactive response on an Intranet cluster

Effectiveness of Distributed Security: We assess in Table 4 the effectiveness of using the micro-firewalls in Intranet for proactive intrusion response. A risk assessment methodology is needed to implement the dynamic policy update, against various threat categories.

This table only provides a qualitative assessment. A comprehensive quantitative evaluation is very much needed through security benchmark experiments to back up these claims. At USC, these experiments are still in progress. A comprehensive DIDS evaluation report is presently under preparation

11. Conclusions

To sum up, we conclude with important lessons learned in bridging wirelined and wireless PKIs and in building the distributed IDS at USC Lab. A few open problems are also identified for further research efforts. The prototype cluster construction and benchmark experiments are still in progress at USC.

Future benchmark work will produce more evidences on the effectiveness in using WPKI and distributed IDS for achieving dynamic network security.

Wireless PKI Architecture: We recommend the Fuzzy Trust Model [19] for risk analysis and the Bridge Trust Propagation Model for implementing the wireless PKI system. This decision leads to higher scalability in the security services.

The clustered WPKI offers also cost-effectiveness in trust path construction and validation. We suggest the cluster architecture to implement the bridge CA system. This will bring high availability and fault tolerance.

WTLS Services Provided: The WPKI should at least implement the Class 2 or 3 WTLS services. The wireless PKI portal must be designed to upgrade the quality of service between mobile clients and wireless service providers. Special software and middleware modules are suggested to manage the WTLS/X.509 cross certificates, CRL, directory/LDAP, and policy configuration.

Table 4 Effectiveness of Distributed Security in Exposed Clusters

Security Threats	Countermeasure Taken and Its Effectiveness
Insider attacks	Fine-grained access control allows only authorized user to communicate with any host in the cluster. Thus insider attacks are better under control.
Denial-of-Service Attacks	Micro-firewalls are distributed, thus they do not act as a single choke point and hence are better in protecting the cluster from flooding attacks from the source.
Trojan programs	Middleware to protect the cluster nodes from trapdoors setup by outside programs. Adding application proxies offers another solution, but it is against the microism.
IP Address spoofing	To reconfigure micro-firewalls to prevent IP spoofing at the node level. Clustered micro-firewalls may use strong authentication to fight against such threats
Probes and Scans	Prevent local scanning and to work with the IDS to block the probes and scans close to their sources of occurrence, lack of proof of its effectiveness at this time.
Unauthorized External access	The node firewalls can be set to disallow any traffic to specific external networks thus blocking this type of threat from initiated from any node in the cluster
Attacks on Cluster infrastructure	Effective if the infrastructure is built in the DMZ against all internal and external attacks and provide fine-grained access control to such infrastructures.

Scalable Directory for Mobile Clients: Reference and replication methods are suggested to enhance the scalability of the directory services for enlarged databases in mobile pervasive applications, because mobile users may increase dramatically in the future. Heavy interactions are expected between mobile users and wireless service providers. Scaling is crucial to support AAA in wireless world. [30].

Interoperability Testing Software: Towards seamless interoperability, we suggest a layered software strategy to develop the testing library. We emphasize inter-domain interoperability to achieve global scalability. These software packages must be benchmark tested in a real pervasive business environment. Joint university and industrial efforts are encouraged to achieve these.

Micro-firewall architecture: The micro-firewall protects an intranet from all attacks. The building blocks of a micro-firewall include the IPchains for packet filtering, extended LIDS for detecting anomalies and intrusions, and extended LogCheck for auditing and logging needed in rule-based intrusion detection and responses at local nodes.

Distributed intrusion detection system: The DIDS is built over micro-firewalls on cluster nodes and on the policy manager in the DMZ. The intrusion detection process is distributed over these hosts inside the cluster. The DIDS works cooperatively with the gateway firewall to yield dynamic security and adaptive intrusion responses.

Security-policy update using XML and RMI: The core of the policy update scheme is the manager host. So far, our major work lies in the Java-based RMI design and XML reporting in security policy changes.

Additional work is needed to test its effectiveness and efficiency. XML key management is a wide-open area for future PKI development [29]. Our continued effort is actively pursuing in this direction.

The USC Internet Security and Pervasive Computing Laboratory was created in 1999. Most of the above security issues are still under intensive investigation. The WPKI architecture is being simulated on a wireless network simulator. A prototype DIDS is running on a Linux PC cluster built in the Lab. Extensive benchmark experiments are still in progress. We will document our research findings in future papers. Visit the Lab web site: <http://andy.usc.edu/trojan/> for updated reports.

Acknowledgements: This presentation is largely based on research performed by my USC research team. In particular, I would like to acknowledge the work of my Ph.D. students: Murali Gangadharan, Yue Chen, Sapon Tanachaiwiwat, and Edward Yang. Some of the work on WPKI and DIDS are still in progress. In addition, my EE 599 students in the Fall 2001 class of *Wireless Internet and Pervasive Computing* had enriched the contents of this presentation through their term projects.

References:

- (1) A & N Associates, Inc. “Federal Bridge Certificate Authority Interoperability Demonstration”, Columbia, MD, Nov.9, 2001
- (2) S. M. Bellovin, “Distributed Firewalls”, *Journal of Login*, Nov 99, pp. 37-39
- (3) M. Carney and B. Loe, “A Comparison of Methods for Implementing Adaptive Security Policies”, *Proc. Of the 7th USENIX Security Symposium*, San Antonio, TX. Jan. 26-29, 1998.
- (4) Certicom, Inc. “The Certicom Trustpoint PKI Portal”, *White Paper*, Hayward, CA. July 2002.
- (5) F. Cohen, “50 Ways To Defeat your Intrusion Detection System”, Cohen and Associates, Livermore, CA. April 2002.

- (6) S. Farrell, "Outlining the Wireless Public Key Infrastructure", Baltimore Tech., July 10, 2001.
- (7) M. Gangadharan and K. Hwang, "Intranet Security with Micro Firewalls and Mobile Agents for Proactive Intrusion Response", *IEEE Int'l Conf. on Computer Networks and Mobile Computing*, Beijing, China October 16-19, 2001.
- (8) U. Hansmann, *Pervasive Computing Handbook*, Springer-Verlag, Berlin, 2001.
- (9) P. Hess and L. Lemire, "Managing Interoperability in Non-Hierarchical PKI", *Proc. of Network and Distributed System Security Symp.*, Internet Society, Reston, VA., 2002
- (10) K. Hwang and Z. Xu, *Scalable Parallel Computing*, McGraw-Hill, San Francisco, 1998.
- (11) K. Hwang, "Internet Security and Firewall Architecture for Reliable E-Commerce and Distributed Computing", Keynote, *IEEE Cluster 2000 Conf.*, Chemnitz, Germany, Nov. 29, 2000.
- (12) K. Hwang and M. Gangadharan, "Micro-Firewalls for Dynamic Security with Distributed Intrusion Detection", *IEEE Int'l Symp. of Network Computing Appl.*, Cambridge, MA. Oct. 8, 2001.
- (13) K. Hwang, Y. Chen, and M. Gangadharan, "Wireless PKI Interoperability for Trust-Based Pervasive Computing, submitted to *IEEE Internet Computing*, April 2002 (under review).
- (14) H. Hsiung, S. Scheurich, And F. Ferrante, " Bridging E-business and Added Trust: Keys to E-business Growth", *IT Professional* , Vol. 3, Issue: 2 , April 2001.
- (15) W. Jansen and T. Karygianis, "Mobile Agent Security", NIST Special Pub. 800-19, Dec. 1998.
- (16) L. Kagal, T. Finin, and A. Joshi, " Trust-based Security in Pervasive Computing Environments", *IEEE Computer*, Vol.34, Issue: 12 , Dec. 2001.
- (17) J. Linn, "Trust Models and Management in PKI", RSA Security Laboratories, Nov.6, 2000.
- (18) R. P. Lippmann, et al, "Evaluating IDS: The DARPA Off-Line Intrusion Detection Evaluation", MIT Lincoln Lab., Lexington, MA. 1999.
- (19) D. W. Manchala, " E-Commerce Trust Metrics and Models", *IEEE Internet Computing*, Vol. 4 Issue 2 , March 2000.
- (20) Miercom, "Intrusion Detection Systems", Lab Testing Report, Princeton, N.J. 08550, April 2001.
- (21) NSS Group, "Public Key Infrastructure", *Group Test Report*, Edition 4, The NSS Group, Cambridshire, England, U.K. Dec. 2001
- (22) R. Perlman, "An Overview of PKI Trust Models ", *IEEE Network*, Vol. 13, Issue 6, Dec. 1999
- (23) M. Petkac and B. Lee. " Security Agility in response to Intrusion Detection", *Proceeding of the Applied Computer Security Associates Conf. 2000*, Louisiana, USA, Dec. 11-15, 2000.
- (24) PKI Forum, "PKI Interoperability Framework", *White Paper*, March 2001.
- (25) W. T. Polk and N. E. Hastings, "Bridge Certification Authorities: Connecting B2B Public Key Infrastructures", NIST, Dec. 2000
- (26) M. J. Ranum, "Experiences Benchmarking Intrusion Detection Systems", *Technical Publication*, NFR Security, Inc. December 2001.
- (27) D. S. Schnackenberg, et al, "Infrastructure for Intrusion Detection and Response", *DARPA Information Survivability Conf. and Exposition (DISCEX)* , Jan. 2000.
- (28) J. Vacca, *Wireless Broadband Networks Handbook: 3G, LMDS, and Wireless Internet*, McGraw-Hill, March 2001.
- (29) VerSign, "XML Key Management, XML Trust Services", White Paper, VerSign, Inc., Moubtain View, CA. 94040, Nov. 2000.
- (30) C. Wang and W. A. Wulf, "Towards a Scalable PKI for E-Commerce System", *IEEE Proc. of the Int'l Workshop on E-Commerce and Web-based Information Systems*, Computer Society, 1998.

Biographical Sketch:

Kai Hwang is a Professor of EECS and Director of Internet Security and Pervasive Computing Lab at the Univ. of Southern California. An IEEE Fellow, he specializes in computer architecture, parallel processing, distributed systems, and pervasive computing. He received the Ph.D. from the Univ. of California at Berkeley.

Dr. Hwang is the founding Editor-in-Chief of the *Journal of Parallel and Distributed Computing*. He has published numerous books and papers. His latest book, *Scalable Parallel Computing* (McGraw-Hill, 1998), covers scalable multiprocessors and multicomputer clusters. He has lectured worldwide and served as a consultant and advisor for IBM, Intel, Fujitsu, MIT Lincoln Lab, Japan's ETL, ETRI in Taiwan, CERN in Holland, and GMD in Germany, etc.

Presently, he leads a research group at USC developing Internet security mechanisms, distributed software RAID for cluster computing, wireless PKI, and distributed IDS for mobile E-Commerce and pervasive computing services. Contact him at: kaihwang@usc.edu or visit his web site: <http://ceng.usc.edu/~kaihwang>