
Copyright 2005 IEEE
Performance Analysis of Unified Enterprise Application Security Framework

Riaz A. Shaikh, Kashif Sharif, Ejaz Ahmed
NUST Institute of Information Technology, Rawalpindi, Pakistan
55riaz@niit.edu.pk, kashif@niit.edu.pk, ejaz@niit.edu.pk

ABSTRACT

Unified Enterprise application security is a new emerging approach for providing protection against application level attacks. Conventional application security approach that consists of embedding security into each critical application leads towards scattered security mechanism that is not only difficult to manage but also creates security loopholes. According to the CSI/FBI computer crime survey report, almost 80% of the security breaches come from authorized users. In this paper, we have worked on the concept of unified security model, which manages all security aspect from a single security window. The basic idea is to keep business functionality separate from security components of the application. Our main focus was on the designing of framework for unified layer which supports single point of policy control, centralize logging mechanism, granular, context aware access control, and independent from any underlying authentication technology and authorization policy.

Keywords: Unified, Enterprise, Application, Security.

1. INTRODUCTION

Our first line of defense is perimeter security, that provides security mostly from external attacks, but it doesn’t provide adequate security against internal attacks. According to the CSI/FBI computer crime survey [1] the total annual losses reported in the 2003 were $201,797,340, in which theft of proprietary information caused the greatest financial loss $70,195,900, unauthorized access by insider caused the financial loss of $406,300. Here the question is, with the rise of improved increasing security technologies, why enterprises are experiencing an increasing rate of malicious activity from both external and internal sources? The answer is that we mainly focus on creating/improving infrastructure security but did not focus on the protection of the application itself in an efficient manner, according to Greg Shipley in [2]

"The emphasis has been on the doors, rather than on what they are protecting. We must become less perimeter-centric and more asset-centric...”

Now we need to look for some new mechanisms in order to protect the critical applications because the attackers are becoming more skillful these days. According to Cert statistics [3]

“In just the last two years the numbers of data security vulnerabilities and incidents have doubled”

New emerging regulations such as Gramm-Leach-Billey act (GLB) [4], Health insurance portability and accountability act (HIPPA) [5], Sarbanes Oxley (Sox) act [6], European Union- Directive 95/46/EC are enforcing pressure on the organizations to protect against attacks. Embedding security into each critical application is one of the solutions in order to overcome some problems, but this kind of approach leads us towards the scattered security mechanism that is difficult to manage. It does not support centralized security control of all application for the entire enterprise, and it will lead to the following challenges [7]

- Each application has its own user management functionalities like Create/Delete users; activate, revoke, or grant permissions for access etc.
- For each application users are forced to remember unique User Id’s and Password that will leads to the selection of weak passwords and/or writing down passwords, making it vulnerable to social engineering. With organizational growth, this factor increases.
- Every time a user joins an enterprise, he/she needs access to multiple applications. Multiple administrators perform Creation of users in these various applications. This creates a loss in productive time.
- Enterprise wide consistent information policy enforcement across all applications become tedious and a massive exercise in itself.
- Logs are distributed and difficult to analysis because each application uses its own conventions and formats [8].
2. UNIFIED APPLICATION SECURITY FRAMEWORK

Authors of [8] proposed the architecture of "Application Defense". The unique thing about this architecture is that it provides complete independence of security issues from the application. Basic Features of this architecture are, 1) Centralized policy management and distributed enforcement, 2) Separate security layer from application, 3) Log all events from entire enterprise in a consistent format 4) Comprehensive audit trials, 5) Incident response of anomalous activity, 6) Confidentiality, 7) Authentication, and 8) role based authorization, granular access control. Conceptually this architecture is divided into three layers, Core layer, Interceptor layer, and Application layer as shown in figure 1.

Core layer consist of, Authentication Server, Certificate Server, Policy Server, Audit Server, Authorization Server, Enterprise Policy Admin Server, and incident response server. Interceptor layer consist of interceptors that are the single access point to the application, and they are placed in front of each application. The main advantage of this architecture is of centralizes policy management and distributed enforcement. Interceptors are also used to log each activity in a centralized database, which is used for real time or off line analysis of anomaly detection. Interceptors are the Frameworks to secure services developed as RMI, CORBA, EJB, HTTP, .NET, XML-RPC, and SOAP applications. Basic requirements of Application Defense (AD) Architecture are;

- AD should handle the activities like policy control, access control and logging at the business level instead of infrastructure level.
- It should operate from single enterprise policy database.
- Logs from different applications like IDS, firewalls, honey pots. Operating systems are in different formats, so AD should maintain all logs for the entire enterprise in a consistent format.
- AD should separate the security layer from the application. It will help in adapting quickly changes in business models.
- AD should not rely on network infrastructure, thus there will be no need to make any changes when infrastructure changes.
- AD should provide reliable real time alerts.

3. INTERCEPTOR MODULE DESIGN

The focus of this research paper is on the designing and development of interceptor layer. As define by ISO 7498/2 the basic information security services are;

1) Confidentiality: refers to the privacy, only sender and receiver will be able to read the message.
2) Integrity: means that data must be arrived at the receiver exactly as it was sent.
3) Authentication: that the receiver is sure of the sender's identity.
4) Access control (Authorization): to allow certain group of people to access certain resources.
5) Non-repudiation: that receiver must be able to prove that a received message came from a specific person.

Interceptor module consists of seven sub modules, which are, authentication, authorization, confidentiality, non-repudiation, integrity, log and session handler modules. In paper [9] we have described this architecture in detail. Here we are going to give very brief overview of each module.

We can authenticate the user or application by different mechanisms like User ID, password, Digital Certificates, Smart Cards, and Biometrics. This module consists of two

---

![Diagram](image.png)

Figure 1. Conceptual layers of Application Security Architecture
basic steps, 1) Identification phase and 2) Validation phase. In identification phase user provides secret information in the form of password or PKI exchanges, on the basis of that information system can validate the user. This authenticate module is independent from any underlying authentication technology like Kerberos, UNIX logging system, LDAP, NT logging system etc.

Several models have been proposed to address the access control requirements [10] like discretionary access control (DAC), Mandatory access control (MAC), Role based access module consist of two basic steps [11], 1) Security lookup phase and 2) Rule enforcement phase. On the basis of identification of user, the system can determine what the caller has access rights? Normally access rights are defines in the enterprise directory such as LDAP directory. When the system gets the access rights of the caller it will compare it with the set of defined rules to permit or deny access to the user. These rules are defined in the form of access control list.

Integrity can be checked by, Message Digest, Checksums or HMAC. Checksums are normally used at transport layers. Since the interceptor is at application layer so we can either use Message Digest or MAC. As compare to simple Message digest, HMAC provide a mechanism that uses the integrity check based on a secret key, so it increases the reliability. So MAC mechanism is recommended for interceptor module.

For checking of non-repudiation we have digital signature mechanism that is based on a public key encryption. Using MD2withRSA, MD5withRSA, SHA1withRSA or SHA1withDSA algorithms can generate digital signature.

To achieve confidentiality we can use either symmetric encryption algorithms such as DES, 3DES, AES etc. or asymmetric encryption algorithms like RSA, blowfish etc.

Log Module works parallel with all previously discussed modules and it is responsible for saving each activity in a centralized logging database in a consistent format. It will help in generating comprehensive real time alerts in a cost effective manner. It will also help in conducting forensic analysis in real time and off line in an easy manner and help in reducing forensic costs by at least two orders of magnitude.

Session handler module is used to handle the unique session of each user and assign unique ID to each user. It also maintain record of session status and make its decision on behalf of that information e.g. if a session will remain idle for certain period of time it will terminate the session.

4. RELATIONSHIP WITH OSI MODEL

Interceptor will cover three layers (application, presentation & session) of OSI model as shown in fig 2. According to OSI security model 7498/2 Authentication is at application layer, Access control is at presentation layer, non-repudiation is at session layer, integrity and confidentiality is at transport and network layer respectively. But as far as interceptor is concerned we are providing security at application level so we put confidentiality at presentation layer because presentation layer is concerned with the syntax and semantics of the information exchanged between two system, and integrity is at session layer because one of the function of session layer is to add checkpoint in a data stream for synchronization, so integrity is needed from one check point to next check point.

5. USE OF ZACCHMAN PHILOSOPHY

Zachman Framework [12] for information system focus on five questions: Why, Who, When, How, What. The same methodology can be use in interceptor module. The process for access control policy criteria, which is shown in figure 3, is a context aware which we will follow
in interceptor. In the first step we identify the application, authenticate that application, when application is authenticated we set the trust level. Trust level is basically depend upon the type of the data of application. Generally we have three trust levels [13], public, private or proprietary, if the application contain public sort of information than normally we require no authentication, secondly if application contain proprietary information than it requires user ID and password and if application contain private data than we need some strong authentication mechanism like digital certificates, smart cards or biometrics. Authentication mechanism will be set depend upon the trust level, and user will be authenticated according to the set authentication mechanism. When user is authenticated we will check user privileges either user is authorized to invoke that application or not. If user has privileges than we will identify what degree of rights the user have to access certain file e.g. read or write etc, after that we will identify that the user trying to access protected data (some file or directory) from certain location is eligible or not. If user is eligible than we verify the time at which user is trying to access the data. If time is valid than user is granted permission to access data either in plain text form or in encrypted form.

6. INITIAL RESULTS AND PERFORMANCE EVALUATION

Current implementation of this unified enterprise application security framework is in progress. We have adopted modular approach and handle each module separately. Our results are based on emulation. We have separately tested three modules for performance evaluation, non-repudiation module, logging module and confidentiality module. Implementations of other modules are currently under way.

In Non-repudiation module, we have taken four digital signature algorithms, 1) MD2withRSA, 2) MD5withRSA 3) SHA1withRSA and 4) SHA1withDSA. We have compared these algorithms on the basis of time taken to create digital signature. We have created the digital signature for different message lengths from range of 100 to 500 bytes for 1000 times and takes the average time, results are shown in figure 5. MD2withRSA takes 23.982ms, MD5withRSA takes 24.327ms, SHA1withRSA takes 24.154ms and SHA1withDSA takes 13.840ms. SHA1withDSA takes lesser time as compare to other algorithms for creating digital signature. So we will recommend SHA1withDSA for interceptor module. Another interesting thing that we found is a what ever the length of message is either 100 bytes or 500 bytes it will approximately take same time to create a digital signature.

Log Module generates the logs into two standard formats first simple format and second XML format. Simple format logs are used to save the logs in a human readable form on the local machine. XML format logs are used to save the log, which contains detail XML structure information to a remote machine. We are following Intrusion Detection Message Exchange format (IDMEF).
Figure 5. Performance of digital signature algorithms

IDMEF [14] is proposed to be a standard data format by Intrusion Detection working group (IDWG) of IETF that automated intrusion detection systems can use to report alerts about events that they look suspicious. We have also compared both formats and found that if log are generating in simple format it will take 3.44ms and in the case of XML format it takes 3.64ms. We have generated both logs 1000 times and take average value of it. Results are shown in figure 6. XML form of logs takes 0.2ms longer time as compare to simple format of logs, but XML format is still recommended in order to achieve consistency and standardization. These logs are generated at application layer, where occurrence of events is not frequent as compare to network layer, so XML format of logs does not create any overhead or bottleneck to the system. Examples of logs generated in XML format and Simple format are given below

XML Format: Sample Output

```xml
<?xml version="1.0" encoding="windows-1252" standalone="no"?>
<!DOCTYPE log SYSTEM "logger.dtd">
<log>
  <record>
    <date>2005-2-08T08:57:55</date>
    <millis>1102478275125</millis>
    <sequence>0</sequence>
    <logger>ASG.Log.ASGLogger</logger>
    <level>SEVERE</level>
    <class>ASG.Log.ASGLogger</class>
    <method>ASGLog</method>
    <thread>10</thread>
    <message>ASG Logging Test Case.</message>
  </record>
</log>
```

Simple Format: Sample Output

Feb 8, 2005 9:00:52 AM ASG.Log.ASGLogger ASGLog SEVERE: ASG Logging Test Case.

Figure 6. Time taken for generating logs

For confidentiality we have given the support of symmetric (DES, 3DES, Blowfish, AES, RC2, RC4) and asymmetric algorithm (RSA). We have also given the support of 5 modes, which are ECB, CBC, OFB, CFB and PCBC with two padding schemes first, no padding second PKCS5 padding. Valid combinations are given in table 1. Default key sizes for DES and Blowfish is 56 bits, 3DES is 112 bits, RC2, RC4 and AES is 128 bits. Default mode is ECB.

Default: ECB = {DES, 3DES, AES, Blowfish, RC2, RC4}

<table>
<thead>
<tr>
<th></th>
<th>No padding</th>
<th>PKCS5 padding</th>
</tr>
</thead>
<tbody>
<tr>
<td>DES/3DES/AES/Blowfish-CBC</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>DES/3DES/AES/Blowfish-CFB</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>DES/3DES/AES/Blowfish-CFB</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>DES/3DES/AES/Blowfish-PCBC</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 1: Valid Combinations

7. RELATED WORK

Authors of [7] proposed a “unified security framework” in which all applications can uses a common security layer for required security processes and uses security services at the centralized level. This will ensure the consistency and increase security management. This unified security layer consists of three distinct components:

1. IT infrastructure -- that contains the applications and processed which required information security.
2. Core security technologies -- It consist of all underlying technologies that provide application level security, encryption, data security, certificate and key management etc.
3. Security processes. -- are the functionalities required for providing authentication, authorization, confidentiality, non-repudiation and higher level of services like single
sign on, user provisioning etc using the core security technologies.

Vormetric Inc. has developed a “Core Guard Security System [15]” that provide a comprehensive solution that protect against, Consequences of exploitable host vulnerabilities, Unauthorized applications and processes, Super user operating outside of their intended mode and purpose, and Direct attacks on stored data. Core Guard data security system prevents attacks like root attack, worms and Trojans, buffer overflow, unintended admin privilege, unauthorized data viewing, audit log tempering, hardware/media theft. It consists of software policy enforcement module (PEMs) installed on hosts that functions as a access point to sensitive information and a security server appliance cluster.

Cerebit Inc, has developed an “InnerGuard System [16]” which comprehensively address the security aspects like authentication, authorization, confidentiality, non-repudiation, integrity and audit at application layer and availability at network layer.

8. CONCLUSION

In unified enterprise application security, interceptor is the core module, which provides all basic information security services like authentication, authorization (access control), integrity, non-repudiation and confidentiality. The basic advantages of this type of framework are, it is able to maintain the logs for each activity in a consistent format. It is a single access point to the application and it works in centralized policy environment. It also provides granular and context aware access control mechanism and it is independent from any underlying authentication and authorization technology.

Organizations related to industry and government such as telecommunication, health care, financial, corporate governance can get benefit by adopting this approach it will help them, in reducing management cost and complexity, reduces the forensic cost, increase reliability, boost customer confidence and compliance to regulatory mandates.

9. REFERENCES


ABOUT THE AUTHORS

**Riaz Ahmed Shaikh** received his Bachelors degree in Computer Engineering from Sir Syed University of Engineering & Technology, Karachi, in 2003. He is currently doing Masters in Information technology with the specialization in networking from NUST Institute of Information Technology, Rawalpindi. His research interests are Network and Application Security.

**Kashif Sharif** completed his Masters degree from NUST Institute of Information Technology, Rawalpindi. Currently, he is serving at NUST Institute of Information Technology, Rawalpindi, as a Lecturer. His areas of interests include QoS in Wireless Networks, and Ad-hoc Networks.

**Ejaz Ahmed** completed his Masters degree from New South Wales University, Sydney, Australia. Currently, he is serving at NUST Institute of Information Technology, Rawalpindi, as a Lecturer. His areas of interests include Information Security, Wireless Ad-hoc Networks.