

# Autonomous Information Unit: Why Making Data Smart Can Also Make Data Secured?

Edward T. Chow  
Enterprise Engineering Division  
Jet Propulsion Laboratory  
California Institute of Technology  
Pasadena, CA 91109  
[edward.chow@jpl.nasa.gov](mailto:edward.chow@jpl.nasa.gov)

## Abstract

*In this paper, we introduce a new fine-grain distributed information protection mechanism which can self-protect, self-discover, self-organize, and self-manage. In our approach, we decompose data into smaller pieces and provide individualized protection. We also provide a policy control mechanism to allow “smart” access control and context based re-assembly of the decomposed data. By combining smart policy with individually protected data, we are able to provide better protection of sensitive information and achieve more flexible access during emergency conditions. As a result, this new fine-grain protection mechanism can enable us to achieve better solutions for problems such as distributed information protection and identity theft.*

## 1. Introduction

The secret agent in the movie Mission Impossible starts by viewing a video tape of the mission assignment. At the end of the video, the voice in the tape says “*This message will self-destruct in five seconds*”. A few seconds later, smoke came out of the tape indicating that the secret mission briefing has been destroyed.

Intelligence information is traditionally protected within a secured facility. Distribution of the intelligence to the field is limited to known targets with the proper clearance. When the intelligence information is distributed to the field, protection can be very difficult. This is the reason why Hollywood uses extreme measures such as smoke from the briefing tape to indicate that the intelligence information will not be taken by any adversary.

Intelligence information protection is going to get more difficult due to the military transformation to net-centric warfare. In the near future, net-centric warfare will provide information superiority for the nation’s military. Programs such as Global Information GRID - Bandwidth Expansion (GIG-BE) and Transformational Communications (TSAT) promise to enable war fighters to get information when and where they need it. The intelligence information protection points will go from a few controlled points to tens of thousands of potentially random points. Many of these points will be on the battlefield, in a moving Humvee, or even in a foxhole. Traditional authentication and authorization-based security mechanisms are not sufficient because they do not support on-demand dynamic access or provide sufficient protection after the intelligence information is downloaded onto the terminals in the field.

Similar problems are happening in civil application of sensitive information. Problem such as identity theft also originated from the wide distribution of sensitive information such as social security and credit card numbers. In order to conduct electronic commerce, these sensitive information usually reside with many on- and off-line merchants. If the information is stolen, the victim often find out after the owner’s credit is ruined.

Bershad’s watchdog implementation [7] provides a mechanism to allow user-definable security function extension on a per-file basis. However, this approach can not control the information security based on the context of the access. Also, in order for watchdog to support modern distributed information access, complicated user extension is needed.

The newer Digital Rights Management (DRM) [1] technology which protects audio/video media or classified information viewing is attempting to resolve the issue of protecting distributed information. However, DRM suffers from problems such as [2] device tethering, lack of super-distribution support, and high complexity of integration. Since most of the DRM implementations today focused on content provider copyright protection, DRM also have the perceived problems of personal privacy intrusion and poorly understood consumer value propositions. The DRM technology provides limited user flexibility; for example, DRM does not understand a doctor's need to access patient medical record due to emergency conditions.

In this paper, we present the architecture and discuss the initial design of a new class of smart information protection mechanism called Autonomous Information Unit (AIU). The AIU provides a fine-grain information dissemination and protection mechanism where information can self-protect and self-organize. An AIU ensures the security and integrity of sensitive information transmitted to the multitude of determined and undetermined end points. The technology provides secured information pre-distribution and allows efficient information access from authorized users whenever and wherever they need it based on the context of the access. In addition, for the problem of misuse of sensitive information such as social security numbers, AIU provides a simple "what you have" solution to the "what you know only" problem.

A number of researchers have addressed the issue of fine-grain data protection. Some focused on methods for efficiently implementing access control lists in database management system [3]. Some address the security problem of sharing data in a GRID computing environment by providing a run time monitoring mechanism [4]. Some focused on model driven security to enable platform independent access control [5]. The AIU addresses the fine-grain protection issue through a policy-based management approach. This enables simple and efficient management by modifying high level policies with changing software implementations.

## 2. Architecture

With today's computing technology, there are applications and data. An application can be executed in the processor but data is just used by application.

The AIU turns data into many tiny, self-contained applications so that they can protect themselves through encryption or other customizable methods.

The AIU is a light weight protected data store. Information is stored in encrypted format within the AIU. Information access is through communication with the AIU. From a typical application perspective, it is just accessing a piece of data on the net. The access process will trigger the execution of the AIU.

AIU provides a level of fine-grain information protection. The data in one AIU may contain only a part of the overall information. The information in the AIU can be automatically composed by partnering AIUs, according to the control policy, the authorization of the user, and the context of access. Simple information ontology may be included in the AIU to support reasoning. A workflow description can also be included to support context processing.

AIU provides fine-grain protection mechanism by putting information in a software enclave. Instead of having one protection mechanism for all sensitive information, this enclave allows information owner to customize different levels of protection for different categories of sensitivities. The AIU allows sensitive information to act differently in different situations. It also allows sensitive information to be distributed to different locations and to come together cohesively when needed.

The AIU also has built in intelligence which can enable AIU to have the following characteristics:

1. Self-Protection:

The information should be able to protect itself. The information should be able to self-destruct when it is no longer needed or when it has been stolen. For example, a terminal containing intelligence photos is taken by the enemy. When the enemy tries to access the photos without proper authorization, the photos should self-destruct in a way that is unrecoverable.

2. Self-Discovery:

The information should be able to understand why it is being accessed. The information should be able to "talk" to other information in order to determine how much access is necessary, if any at all. If the context is not correct then it should deny access. For example, when a bank tries to use an AIU that contains the social security number of an applicant, the AIU should be able to contact its

owner to determine if the access is allowed. The permission for use of the social security number is composed of both the number (what you know) as well as the AIU with the permission from the owner (what you have).

3. Self-Organization:

The information should be able to obtain other information to complete the request needed by the user. One way to protect intelligence is to not store the complete information in one location. The information should be able to coordinate with a group of other data repositories in order to obtain the complete information needed by the user. For example, only a portion of the intelligence photo pixels is provided to the Special Forces unit before they move into position. When they arrived at the location, per their new GPS location value, the rest of the pixels are automatically downloaded to complete the picture.

4. Self-Management:

The information should be able to manage itself according to the originators' policies. The information should be able to automatically communicate with the originators of the information and update the policies. For example, the content owner should be able to set the policy associated with an intelligence photo's update rate to 12 hours and the information should try to retrieve updates every 12 hours. If the AIU can not automatically update, then it should be able to check for updates when it's been accessed.

parts of the metadata and policy rules may be visible when communicating with the AIU. The ASPM is also responsible for providing location services between related AIUs. After the AIU is formatted, it will be distributed to a target terminal or AIU cache. The AIU Management Service (AMS) in the AIU cache is a special type of AIU which is always running in the background. The AMS provides remote data management, distributed search, and activity scheduling functions. It can support distributed search at a metadata level. It is also responsible for waking up the AIUs for management operations such as automatic updates according to pre-defined policies of the AIU and ASPM.

The AIU caches are used to store AIUs in distributed locations. The advantage of this approach is that high speed information access can be enabled for both the high-bandwidth network as well as the long-latency satellite network. For example, the white color AIUs in Figure 1 are distributed in different locations. If the white AIU in the Humvee requests more information to complete the intelligence, it may need to talk to other white AIUs to obtain that information. This is not a problem in the high-speed GIG-BE environment. However, in the TSAT environment with long latencies, the back and forth interactions between geographically distanced AIUs can create long delays for the user. The local AIU cache will be able to solve this problem.

The block diagram of an AIU is shown in Figure 1. The intelligence information, metadata, and policy rules are stored in sealed storage. The Storage Controller is responsible for the access of the sealed storage based on the level of authorization. The Policy Manager is responsible for the operations of the AIU. Based on the access context, the Policy Manager can control the Inference Engine either to make local decisions or to communicate with other AIUs through the Network Interface. The Inference Engine is used to process the rules used for self-protection, self-discovery, self-organization, and self-management.

The AIU approach has many advantages over the traditional data protection mechanisms:

1. Security:

AIU can provide distributed fine-grain data protection. AIU can perform access right validation from multiple remote sources. AIU can grant or deny access based on context of the access and pre-defined policy. AIU can also be customized to provide different levels of

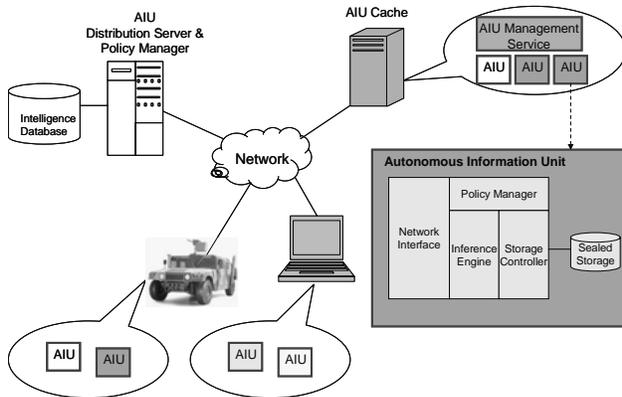


Figure 1: AIU System Block Diagram

As depicted in Figure 1, the AIU system contains several elements. The AIU Distribution Server and Policy Manager (ASPM) are responsible for formatting the AIU by attaching the AIU metadata and policy rules. Depending on the level of authorization, some

protection for different type of information. It can use different encryption strength depends on the level of protections needed.

2. Flexibility:

Traditional role-based access control requires an administrator's involvement to change or revoke user access rights. This approach does not meet the net-centric operations requirement of allowing access whenever and wherever data is needed. The AIU approach allows true, dynamic access determined by the access context.

3. Performance:

AIU's fine-grain protection mechanism allows parts of the information to be pre-distributed without threatening the security of the complete information. This allows higher performance access for large data file due to the local caching of partial contents.

4. Management:

AIU provides enterprise a uniform management mechanism. The AIU supports policy-based management. This allows inherent security controlled by enterprise policy. Since many businesses do not have IT support in remote offices, managing a complicated remote database system can be problematic. AIU does not require a database system in remote locations. Managing the security and backup operations can all be done remotely through policy updates.

### 3. Implementation

There could be a number of different implementation approaches for the AIU architecture. Our approach is to provide a solution to address the following considerations:

1. Protection:

The AIU can have many advantages but it also creates some concerns: Can the AIU be trusted? Can AIU communication be observed by hackers? Can AIU turn into a virus and destroy other data in the system?

Our solution to the protection problem is based on a number of new technologies. It is estimated that there are over 5 million computers equipped with Trusted Platform Module (TPM) today. The Trusted Computing Group (TCG) industry initiative designed the TPM as a hardware device that can not be tempered. It creates a source of trust. The Microsoft Next Generation Secured Computing Base (NGSCB) can use the TCG

standards to create trusted protected computing "Nexus". Some elements of the NGSCB will be available in the soon to be released Microsoft Windows Vista operating system. A Nexus Computing Agent (NCA) is an isolated computing enclave which has (virtually) its own CPU, keyboard, and screen. It offers a parallel execution environment to the traditional Windows kernel-mode and user-mode stacks. It creates a new environment that runs alongside the operating system, not underneath it. Since it is isolated, it can not destroy the rest of the system. The cell phone industry is also adopting the TCG technology. Within the next few years, there may be hundreds of millions of computers and cell phones protected by this type of trusted computing technology. Many of the net-centric platforms will be protected by the same technology.

According to Microsoft, a Nexus-aware PC will offer four categories of new security features:

- **Strong process isolation.** Users can isolate pages of main memory so that each Nexus-aware application can be assured that it is not modified or observed by any other application or even the operating system.
- **Sealed storage.** A Nexus-aware application or module can mandate that the information be accessible only by itself or by a set of other trusted components that can be identified in a cryptographically secure manner.
- **Secure path to and from the user.** Secure channels allow data to move safely from the keyboard or mouse to Nexus-aware applications and from Nexus-aware applications to a region of the screen.
- **Attestation.** Software can digitally sign or otherwise attest to a piece of data and thus assure the recipient that the data was constructed by an un-forgeable, cryptographically identified, trusted software stack.

Our approach is to design the AIU as a small NCA. Information access is through communication with the NCA. The access process will trigger the AIU NCA to be executed. The execution of the AIU will not affect the system because it is a NCA.

The result of the trusted computing environment is a revolutionary hardware/software

platform that can provide assurance that protected environments are correctly invoked. The solution guarantees the isolation so that protected environment can not destroy the rest of the system. The environments also provide data protection, preventing storage and memory inputs/outputs from being observed or compromised by any unauthorized software running on the platform. These capabilities enable us to create an Autonomous Information Unit (AIU) that can be trusted.

## 2. Data Decomposition and Reassembly:

The AIU can enable fine-grain data security. Our implementation needs to address the process of decompose data so that it can provide the best security protection and can be efficiently reassembled with less computing and communications overhead.

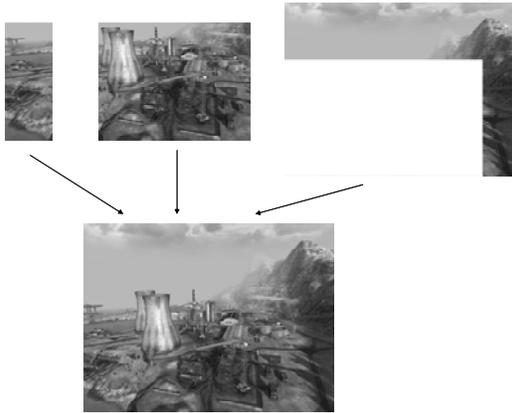


Figure 2: AIU data decomposition and reassembly

We divide data decomposition into two categories that can be supported by the AIU; simple and complex decompositions. For the simple decomposition, data is decomposed in a structured way where there are simple indexes for reassembling the data together. An example of this is shown in Figure 2 where a portion of an image is decomposed into a separate AIU. The AIU metadata will contain data fields such as starting point, dimensions, size, and etc. This allows reassembly by combining AIU data together according to these data fields. The AIU metadata also include references to other related AIUs which contains other parts of the decomposed data. These references are used to lookup other related AIUs in ASPM or AMS.

Complex decomposition includes two additional functions than the simple decomposition. The metadata can contain a custom defined function for reassembly. The AIU will apply the function when reassemble the data together. The metadata can also contain a context description where the reassembly of the AIU data can result in different final information depending on the current access context. The context description is represented in an executable policy format.

## 3. Policy Manager:

The Policy Manager is the overall AIU controller. In order to reduce the overhead of the AIU, the Policy Manager and the Inference Engine cannot be large and slow.

Our design of the Policy Manager and the Inference Engine is based on the JPL SHINE [6] technology. The JPL SHINE inference engine has been independently evaluated to outperform all commercial products by at least two orders of magnitude. It provides a 100X to 1000X improvement in inference speed and up to 10,000X reduction in execution environment size. The SHINE inference engine can executes over 230,000,000 rules per second on a standard 3.8 GHz Windows XP desktop PC and more than 33,000,000 rules per second on flight hardware. It provides supercomputer performance on conventional hardware. A self-executing policy engine with 100 rules can be compiled into 17 Kbytes based on SHINE technology.

When an AIU combines the data from another AIU, the policies of both AIUs need to be combined. In our design, the combining AIU has priority over the combined AIU. The combined policy uses the minimum scope accepted by both AIU's policies.

## 4. Communications:

When an AIU communicate with other AIUs or AIU cache, it must be able to communicate through firewalls.

Our design of the AIU is based on Web Services model. AIU can be called as a Web Services. The called AIU will conduct a secure Web Services conversation with the calling application or AIU. The called AIU may initiate a conversation with other AIUs to assemble the complete information requested. State is

maintained during the conversation process.

A typical simple operation can start with an application makes a Web Services call to an AIU. The AIU will perform authentication and authorization through either local lookup or remote operations. The Policy Manager is involved in this process to enforce the policy defined by the content owner. Based on the information requested, the AIU may determine to request information from other AIUs through similar Web Services calls. The Policy Manager is also involved in this process to determine how data will be reassembled.

#### 4. Application Examples

We present three application scenarios for AIU in this section.

##### 1. Intelligence Data Assembly:

In this scenario an intelligence photo of a target site for Special Forces can be decomposed into several pieces and stored in several AIUs. Some of the AIUs can be carried by the Special Forces and some can be kept in the AIU cache. When the Special Forces arrive at the target location, the soldier can call the entry point mission AIU which will assemble the rest of the intelligence photo together from local and remote locations. The advantages of this approach include better bandwidth efficiency and better intelligence protection before the mission.

##### 2. Personal Information Validation:

Another scenario is the use of personal information such as social security number and credit number. Assuming these numbers are stored in AIUs with personal user defined policies. These policies can include call back request to the owner of these personal information. The call back can be in real time such as cell phone validation or non real time mode through email. The owner has to accept the call back and authorize the AIU for the merchant to have the true authorization to use the information. The advantage of this approach is that the personal information AIU can be distributed without worry of been misused. Whenever the personal information AIU is used there will be a record of the use by both the merchant as well as the owner of the information.

##### 3. Medical Record Release:

One of the problem in electronic medical record is the worry of been released to unauthorized person. At the same time, there is also the worry of not being able to obtain the record by medical personnel during emergency condition where the owner is not able to authorize the release. In this scenario assuming the medical personnel is trying to access the medical record of a patient. The Medical Record AIU can include a policy which says that if the patient is not able to respond and the request comes from a authorized medical personnel then the access can be allowed.

#### 5. Conclusion

In this paper we presented the architecture of a new distributed information protection mechanism. The AIU mechanism put data into a fine-grain enclave and provide individualized protection based on owner defined policies. By combining smart policy with the protected data, we are able to provide better protection of sensitive information and achieve more flexible access during emergency conditions.

#### 6. References

- [1] Hartung, F. and Ramme, F., "Digital rights management and watermarking of multimedia content for m-commerce applications", IEEE Communications Magazine, Nov. 2000.
- [2] Rosenblatt, B., "Gaps in Existing DRM Technology", <http://www.drmwatch.com>, Nov. 2003.
- [3] Kelter, U., "Discretionary Access Controls in a High-Performance Object Management System," IEEE Symposium on Security and Privacy, 1991.
- [4] Butt, A.R. Adabala, S. Kapadia, N.H. Figueiredo, R. Fortes, J.A.B., "Fine-grain access control for securing shared resources in computational grids", Parallel and Distributed Processing Symposium, April 2002.
- [5] Burt, C.C. Bryant, B.R. Raje, R.R. Olson, A. Auguston, M., "Model driven security: unification of authorization models for fine-grain access control", IEEE Enterprise Distributed Object Computing Conference, Sept. 2003.
- [6] James, M. and Aldiwan, W., "Autonomous tools and techniques for reducing operational and maintenance costs in space", IEEE Aerospace Conference, March, 1999.
- [7] Bershad, B. and Pinkerton, C., "Watchdogs: Extending the UNIX File System," Proceedings of the 1988 Winter USENIX Conference, Feb. 1988, pp. 267-276.

---

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.