Formal threat descriptions for enhancing governmental risk assessment

Andreas Ekelhart
Secure Business Austria
Favoritenstraße 16/2
Vienna, Austria
aekelhart@securityresearch.at

Stefan Fenz
Secure Business Austria
Favoritenstraße 16/2
Vienna, Austria
sfenz@securityresearch.at

Thomas Neubauer
Secure Business Austria
Favoritenstraße 16/2
Vienna, Austria
tneubauer@securityresearch.at

Edgar Weippl
Vienna University of Technology
Favoritenstraße 9-11
Vienna, Austria
weippl@ifs.tuwien.ac.at

ABSTRACT

Compared to the last decades, we have recently seen more and more governmental applications which are provided via the Internet directly to the citizens. Due to the long history of IT systems in the governmental sector and the connection of these legacy systems to newer technologies, most governmental institutions are faced with a heterogeneous IT environment. More and more governmental duties and responsibilities rely solely on IT systems which have to be highly dependable to ensure the proper operation of these governmental services. An increasing amount of software vulnerabilities and the generally heightened physical threat level due to terror attacks and natural disasters demand for a holistic IT security approach which captures, manages, and secures the entire governmental IT infrastructure. Our contribution is (1) a novel inventory solution, (2) a mechanism to embed the virtual IT infrastructure data into a physical model provided by our security ontology, and (3) a methodology to automatically identify threatened assets and to reason on the current security status based on formal threat definitions taking software configurations and physical locations into account. A prototypical implementation of the aforementioned concepts shows how these concepts help governmental institutions to secure their IT infrastructure in a holistic and systematic way to fortify their IT systems in an appropriate way against current and future threats.

Categories and Subject Descriptors

C.2.0 [Computer-Communication Networks]: General— security and protection; C.2.3 [Computer-Communication Networks]: Network Operations—network monitoring

ICEGOV 2007, December 10-13, 2007, Macao

General Terms

Security, Management

Keywords

inventory, security ontology, formal threat descriptions

1. INTRODUCTION

The past years have seen increasingly rapid advances in running governmental processes more effciently by the usage of information technology. Today, a growing number of governmental applications are provided via the Internet directly to the citizens (e.g., online tax computation). One of the driving forces for this development in Europe is the strategic policy framework i2010: Information Society and the media working towards growth and jobs of the European Union [2], that promotes the application of web-based public services in the years up to 2010. Due to the long history of IT systems in the governmental sector and the connection of these legacy systems to newer technologies, most governmental institutions are faced with a heterogeneous IT environment.

Nowadays, governmental duties and responsibilities rely solely on IT systems which have to be highly dependable to ensure the proper operation of these governmental services. An increasing amount of software vulnerabilities according to CERT/CC statistics 1988-2006 [1], the number of reported software vulnerabilities doubled in the last three years and the generally heightened physical threat level due to terror attacks and natural disasters demand for a holistic IT security approach which captures, manages, and secures the entire governmental IT infrastructure.

In this paper we present: (1) A novel inventory solution, which is able to capture the device data (e.g., operating system and IP address) independent of the used operating system. A plug-in mechanism enables the usage of third party products such as Nmap [7] or Microsoft Windows Server Update Services [8] to gather the device data in the most granular and reliable form.

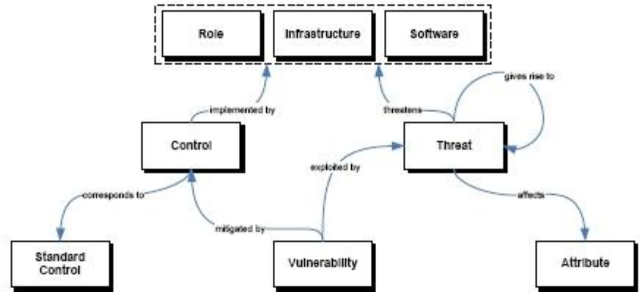


Figure 1: Security Relationships

(2) A mechanism to embed the virtual IT infrastructure data into a physical model provided by the Security Ontology[1]. (3) A methodology to automatically identify threatened assets and to reason on the current security status based on formal threat definitions taking software configurations and physical locations into account. This approach helps governmental institutions in planning and implementing a holistic IT security approach by providing comprehensive information about threat types and their impact on the governmental assets.

2. DATA MODEL

The automated identification of threatened assets requires a formal and machine-readable threat definition, namely an ontology. Therefore, we proposed the Security Ontology [3] classification which is based on the security relationship model presented in [9] (see Figure 1). Threats, vulnerabilities, controls and their implementation (safeguards) are the pivotal elements: a threat represents, through an existing vulnerability, any potential danger to governmental assets and is initiated by a threat agent. To pose a risk to an organization, a threat has to exploit a vulnerability, via a physical, technical or administrative weakness, and cause damage to defined assets. Controls have to be put into place to mitigate an identified vulnerability and to protect the corresponding assets by either preventive, corrective or detective measures.

Threat modeling on such a highly granular level ensures that corresponding vulnerabilities and threatened software, role, and infrastructure elements are connected in the most direct and meaningful way. Threat mitigation can be only achieved by mitigating the corresponding vulnerabilities through the implementation of connected controls. Threats such as fire, asset destruction, and data loss are latent and need an exploitable vulnerability to become destructive. Each control within the Security Ontology corresponds to an information standard control coming from the international ISO 27001 [6] standard or the German Baseline Protection Manual. The benefit of this approach is two-fold: on the one hand it ensures that best practices are incorporated within the Security Ontology and on the other hand it enables the support for a potential certification regarding these standards.

After the definition of abstract security relationships and the enrichment with concrete threat and vulnerability information it is necessary to relate this knowledge with data about the governmental assets.

3. DATA COLLECTION

As depicted in Figure 1, we group the governmental assets into three sections: (1) roles, which are associated with human beings, (2) infrastructure, mapping physical assets, and (3) software, representing concrete data sets as well as software packages such as operating systems and ofice suites. While roles and most infrastructure elements are entered manually, we propose an inventory solution for the software and IT-related infrastructure elements which is able to capture the device data automatically (operating system, IP address, patch level, etc.) independent of the used operating system. Collecting such detailed device data enables, in the case of software-related threats, the mapping of software vulnerabilities on the current IT infrastructure, to visualize the threatened systems.

We propose a novel inventory solution which utilizes agent- and non-agent-based methods coming form third party providers. The execution parameters of each sensor (third party product) are configured in a central XML file and the output is also written in a XML structure. If the inventory solution does not run for the first time, it loads the XML file, containing IT infrastructure inventory data from former scans. The net-inventory step utilizes network tools such as Nmap, Ping or ARPPing to gather amongst others device data, IP addresses and OS fingerprints. The main purpose of this step is to gather as much relevant data as possible about IT infrastructure elements to create an up-to-date network topology documentation. The host-inventory step incorporates the output of several third party products to gather detailed device data including installed software packages and OS version. Depending on the IT infrastructure, most governmental institutions have already some kind of software inventory solutions in place. Windows Server Update Services, Red Hat Network, and Apple Remote Management are only a few examples for proprietary solutions which manage the software status of connected clients. Our inventory solution aims at tapping these systems to use their data for risk analysis.

By combining the inventory solution with the Security Ontology approach we are able to embed the gathered IT infrastructure data into a physical model (comprising buildings with corresponding floors, rooms and further infrastructure elements). By assigning physical locations to the virtual infrastructure data we are able to secure the entire IT infrastructure in the most holistic way, because we are taking the physical as well as the software security into account. As a physical control example, access to rooms containing critical server systems should always be monitored and restricted by proper access control systems. On the software security side, a newly released vulnerability description can be parsed and stored in the ontological knowledge base and subsequently affected systems in the organization can be automatically reasoned.

4. RISK ASSESSMENT

The risk assessment is subdivided into two steps: (1) the threat impact assessment where the potential damage of physical threats such as fire and vandalism is considered, and (2) the software vulnerability impact assessment, where data of software vulnerabilities is mapped to existing IT systems. As motivated in the previous sections, automatic software and hardware-related risk assessment support can be conducted by reasoning on established knowledge, hence we identify two pivotal components: a knowledge base and corresponding rules. Due to its high semantic potential we decided for an OWL-based

[1] Security Ontology: securityontology.securityresearch.at, last access: 19 October 2007

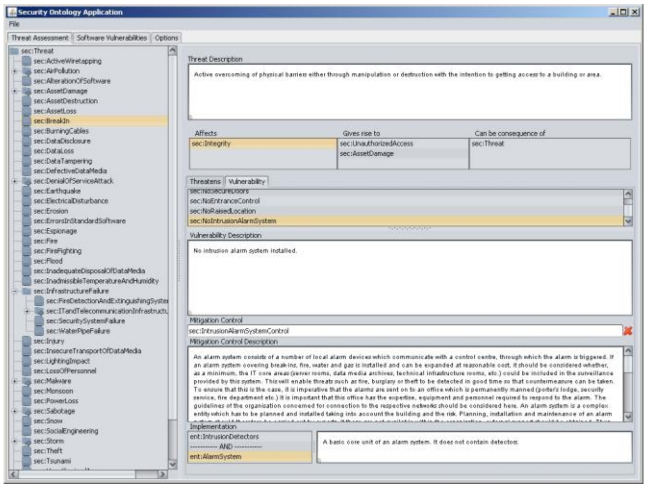
knowledge store. One of the core modules of our solution is a semantic engine, based on Jena [5] and Protege packages which ofiers the methods for manipulating and querying OWL-based data. The interface is split into two main areas, namely threat impact assessment and software vulnerability impact assessment, reecting our interrelated research approaches.

4.1 Threat Impact Assessment

Risk management uses risk analysis as a tool to identify vulnerabilities and threats and to assess the possible damage to determine where to implement security safeguards. Companies demand for cost efiective, relevant and timely security implementations, however, security can be quite complex.

The current prototype version aims at supporting the persons responsible for security, mainly in the following steps of a risk analysis: identifying assets and their values, identifying vulnerabilities and threats and ofiering corresponding countermeasures. In addition, automatic evaluation of existing security measures, based on formal security requirement definitions, can be conducted and provides further help to establish and keep the required security level. Figure 2 shows a screenshot of the risk analysis support interface. The threat tree, located at the left hand side, is the starting point for any risk analysis in our model. The underlying information is gained from the OWL-based knowledge base; details on the structure and modeling process can be found in Section 2. Each of the items under the sec:Threat root element represents a possible threat to the organization and has to be taken into account in a holistic risk analysis. By selecting a threat from the tree representation, the right area is populated with valuable information. On top a threat description is provided in natural language. Below, afiected security attributes are displayed. Furthermore, a threat can be a consequence of other threats (e.g., unauthorized access can be the result of a break in or missing key management) and can itself potentiate other threats (e.g., break-in gives rise to unauthorized access or asset damage). These interdependencies span information-rich threat trees which provide a comprehensive resource for risk analysis. The Threatens area shows the threatened classes which are connected with the selected threat (e.g., break-in threatens ent:Building, ent:Door, and ent:Window). Referring to Section 3 the organization has been inventoried beforehand (under defined classes), and thus concrete threatened company instances are given implicitly.

At this point, the user has detailed knowledge about possible threats and identified endangered systems (including data about asset costs, delivery time, and location) within his organization. By means of this information she can prioritize threats and subsequently start to search for safeguards and develop an implementation strategy. Our prototype also addresses these steps by another major area, namely the vul nerability view (compare Figure 2). For each threat highly granular vulnerabilities, which a threat could exploit, have been defined and modeled in the ontology. On example of break-in, the assigned vulnerabilities are No Secure Windows, No Secure Doors, No Entrance Control, No Raised Location, and No Intrusion Alarm System.



Threat Description: Active overcoming of physical barriers either through manipulation or destruction with the intention to getting access to a building or area.

Affects	Gives rise to	Can be consequence of
sec:Integrity	sec:UnauthorizedAccess sec:AssetDamage	sec:Threat

Vulnerability:
sec:NoSecureDoors
sec:NoEntranceControl
sec:NoRaisedLocation
sec:NoIntrusionAlarmSystem

Vulnerability Description: No intrusion alarm system installed.

Mitigation Control: sec:IntrusionAlarmSystemControl

Mitigation Control Description: An alarm system consists of a number of local alarm devices which communicate with a control centre, through which the alarm is biggered. If an alarm system covering break-ins, fire, water and gas is installed and can be expanded at reasonable cost, it should be considered whether, as a minimum, the IT core areas (server rooms, data media archives, technical infrastructure rooms, etc.) could be included in the surveillance provided by this system. This will enable threats such as fire, burglary or theft to be detected in good time so that countermeasure can be taken. To ensure that this is the case, it is imperative that the alarms are sent on to an office which is permanently manned (porter's lodge, security service, fire department etc.) It is important that this office has the expertise, equipment and personnel required to respond to the alarm. The guidelines of the organisation concerned for connection to the respective networks should be considered here. An alarm system is a complex entity which has to be planned and installed taking into account the building and the risk. Planning, installation and maintenance of an alarm

Implementation:
ent:IntrusionDetectors
---------- AND ----------
ent:AlarmSystem

A basic core unit of an alarm system. It does not contain detectors.

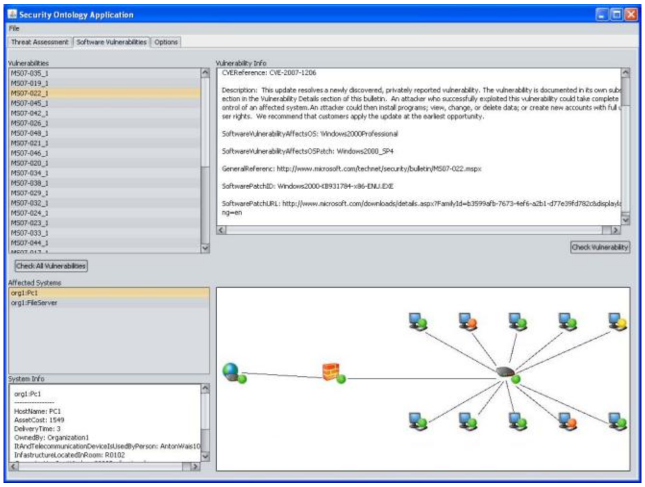
Figure 2: Threat impact assessment view

A description of each vulnerability in natural language complements the vulnerability presentation. For each of the vulnerabilities a mitigation control is assigned, thus implementing a control closes a vulnerability. To enhance understanding, each control is enriched by a natural language description. With these functions in place, a user knows exactly how to protect his organization from specific threats: mitigating vulnerabilities by implementing recommended controls. To facilitate the aspect of automatic compliance checks regarding our defined mitigation controls, each control further incorporates formal implementation descriptions. The implementation area in Figure 2 shows the actual implementation measures for a control. Referring to our example, the threat Break-in exploits the vulnerability No Intrusion Alarm System which could be mitigated by the installation of an intrusion alarm system and an intrusion detector (motion detector, glass break sensor, or heat detector) in every room. The underlying formal descriptions can be executed as rules against the organizations concrete modeled environment to identify which parts of the building are in compliance.

4.2 Software Vulnerability Impact Assessment

The output of the inventory solution, as described in Section 3, is used to map semantically enriched vulnerability information automatically on the ontological stored IT infrastructure data to visualize vulnerable systems. Compared to the threat impact assessment, the software vulnerability impact assessment is designed for a real-time analysis.

Figure 3 shows a prototype screenshot of the software related view. In the list area on the left hand side, software related vulnerability messages are listed which were initially collected by the vulnerability message collector software. This program subscribes to RSS feeds and thereby downloads vulnerability definitions the moment they are published. After the transformation phase the vulnerabil ity definitions are stored as XML files.



Vulnerabilities

MS07-035_1
MS07-019_1
MS07-022_1
MS07-045_1
MS07-042_1
MS07-026_1
MS07-048_1
MS07-021_1
MS07-046_1
MS07-020_1
MS07-034_1
MS07-038_1
MS07-029_1
MS07-032_1
MS07-024_1
MS07-023_1
MS07-033_1
MS07-044_1

Vulnerability Info

CVEReference: CVE-2007-1206

Description: This update resolves a newly discovered, privately reported vulnerability. The vulnerability is documented in its own subs
ection in the Vulnerability Details section of this bulletin. An attacker who successfully exploited this vulnerability could take complete
ontrol of an affected system.An attacker could then install programs; view, change, or delete data; or create new accounts with full
ser rights. We recommend that customers apply the update at the earliest opportunity.

SoftwareVulnerabilityAffectsOS: Windows2000Professional

SoftwareVulnerabilityAffectsOSPatch: Windows2000_SP4

GeneralReferenc: http://www.microsoft.com/technet/security/bulletin/MS07-022.mspx

SoftwarePatchID: Windows2000-KB931784-x86-ENU.EXE

SoftwarePatchURL: http://www.microsoft.com/downloads/details.aspx?FamilyId=b3599afb-7673-4ef6-a2b1-d77e39fd782c&displayla
ng=en

Affected Systems

org1:Pc1
org1:FileServer

System Info

org1:Pc1

HostName: PC1
AssetCost: 1549
DeliveryTime: 3
OwnedBy: Organization1
ItAndTelecommunicationDeviceIsUsedByPerson: AntonWais10
InfastructureLocatedInRoom: R0102

Figure 3: Software vulnerability view

By utilizing a file watcher, newly created vulnerability definitions are detected and subsequently imported into the ontological knowledge base. As a consequence to changes in the vulnerability repository, newly arrived vulnerabilities are immediately evaluated against the organization's infrastructure. The following scenario details this process: throughout the example we refer to our test environment which we set up and used during the development phase. This environment consists of technical elements such as servers, or client systems and is located in an ofice building spanning two floors.

On 26 June 2007 Microsoft updated the critical vulnerability message MS07-22 which informs customers about a vulnerability in theWindows kernel which could allow elevation of privilege. At this point our process starts with the vulnerability collector receiving this information in a feed message. After parsing the information the data is imported into the semantic vulnerability store. A file watcher initiates the evaluation functions which rely on SPARQL defined queries and a vulnerability rule set. An example query would result in a listing of matching systems, namely org1:Pc1 and org1:FileServer (compare Figure 3). In the proposed ontological vulnerability storage format, solutions are integrated if existing (e.g., software patch IDs) and thereby as a next step, already patched systems can be discarded from the identified system list. Now, the final list of potentially vulnerable systems is presented to the organization's IT administration. By selecting a vulnerability message, detailed information on the vulnerability as well as on afiected systems is displayed. As Figure 3 also illustrates, one effective way for presentation are network plans including signs to visualize problems. We have chosen trafic lights for our initial prototype as the colors red, yellow and green are intuitively understood. Systems ashing red draw attention and signalize danger, yellow denotes an uncertain state with potential problems and has to be investigated and green indicates a healthy system. At one glance the overall organization's IT network can be monitored, which is also an attractive option for management purposes. Passing over the system symbols in the graphic calls up information on the device and the reason for the set state (e.g., the vulnerability description in case of a red state). The responsible individual has now comprehensive information on hand (including solutions such as the download location of the patch) to judge on the vulnerability's impact and the necessary consequent steps.

5. CONCLUSION

In this paper we presented a novel inventory solution for IT systems, a mechanism to embed the virtual IT infras tructure data into a physical model, and a methodology to automatically identify threatened assets based on formal threat definitions taking software configurations and physical locations into account. The multitude of threats can be considered systematically in governmental operation security to mitigate them to the lowest possible level. The systematic approach also includes mitigation solutions for the corresponding vulnerabilities and enables the user by utilizing best-practices to secure the IT environment in a systematic and established form. Further research will address the ontological mapping of elemental mitigation controls to the Common Criteria and ISO27001 ontologies as proposed in [4] to allow the user to investigate the organization's security status in regard to these security standards. This approach supports a potential standard certification, aiming at lowering the financial costs and time required for the certification procedure.

6. REFERENCES

[1] CERT/CC. Cert/cc statistics 1988-2006. http://www.cert.org/stats/, January 2007.

[2] Commission of the European Communities. Communication from the commission tot he council, the european parliament, the european economic and social committee and the committee oft he regions ‘i2010 – a european information society for growth and employment‘. COM(2005) 229 final, June 2005

[3] A. Ekelhart, S. Fenz, M. Klemen, and E. Weippl. Security ontologies: Improving quantitative risk analysis. In *40th Hawaii International Conference on System Sciences (HICSS'07)*, pages 156-162, 2007. IEEE Computer Society.

[4] S. Fenz, G. Goluch, A. Ekelhart, and E. Weippl. Information Security Fortication by Ontological Mapping of the ISO/IEC 27001 Standard. In *13th Pacific Rim International Symposium on Dependable Computing, PRDC2007*. IEEE Computer Society, December 2007.

[5] Hewlett-Packard Development Company. JENA semantic web framework. http://jena.sourceforge.net/, August 2007.

[6] International Organization for Standardization and International Electrotechnical Commission. Iso/iec 27001:2005, information technology – security techniques - information security management systems requirements.

[7] I. LLC. Nmap security scanner. http://insecure.org/nmap/, June 2007.

[8] Microsoft Corporation. Windows server update services. http://www.microsoft.com/germany/windowsserver2003/ technologien/updateservices/default.mspx, June 2007.

[9] NIST. An introduction to computer security - the nist handbook. Technical report, NIST (National Institute of Standardsand Technology), October 1995. Special Publication 800-12.