A NEW VULNERABILITY TAXONOMY BASED ON PRIVILEGE ESCALATION

Zhang Yongzheng
National Computer Information Content Security Key Laboratory,Harbin Institute of Technology, No.92, West Da-Zhi Street, Harbin, China

Yun Xiaochun
National Computer Information Content Security Key Laboratory,Harbin Institute of Technology, No.92, West Da-Zhi Street, Harbin, China

Keywords: Security risk assessment; Security vulnerability; Vulnerability taxonomy; Privilege escalation

Abstract: Computer security vulnerabilities badly compromise the system security. To profoundly understand the causes of known vulnerabilities and prevent them, this paper develops a new taxonomic character, and then integrates a privilege-escalating based vulnerability taxonomy with multidimensional quantitative attribute. This taxonomy greatly contributes to further researches of security risk assessment of computer system.

1 INTRODUCTION

With the same rapid growth as the Internet itself, malicious code, viruses, Trojan attacks and other ‘hacks’ have flourished. But most of these attacks are arisen by security vulnerabilities which badly compromise the system security. So the systematical research of vulnerability taxonomy is of guidance significance for profound understanding the causes of the known vulnerabilities, for further preventing them, and for detecting the new vulnerabilities. Also, this research gives a helping hand in warning software developers and users against the same errors again. Now, vulnerability taxonomy is mainly applied to the fields of security evaluation, security audit, and IDS of computer. However, the taxonomy in this paper is specially designed for security risk evaluation of computer system.

Many research organizations and researchers are engaged in this work and gain some achievements. On the whole, more famous and effective taxonomies are mainly the following: Aslam’s Taxonomy (Aslam,1995;Aslam et al.,1996), M.Bishop’s Taxonomy (Bishop and Bailey, 1996), E.Knight’s Taxonomy (Knight and Hartley, 2000) and K.Jiwnani’s Taxonomy (Jiwnani and Zelkowitz, 2002). However, there are the same defects in these taxonomies: (1) Lack or roughness of quantitation. They can’t reflect the harm degree of vulnerabilities on finer granularity, so that the compromise of vulnerabilities can’t come to users’ knowledge. (2) Can’t reflect the relationship of privilege escalations aroused by vulnerabilities. In fact, a vulnerability whose harm is low on the surface possibly leads to a series of privilege escalations, so as to bring the serious harm to system. Moreover, these flaws are usually ignored by users. (3) Can’t express the probability of successful exploiting flaws. It is said that the simpler attack tools are, the higher the successful rate of exploitations is.

The above defects are what we anticipate improving in practice. L.D.Wang presents the idea of privilege escalation and gives a framework of taxonomy in (Wang, 2002). Based on it, this paper introduces part concepts and further improves. Thus this paper develops a new taxonomic character, and then integrates a privilege-escalating based vulnerability taxonomy with multidimensional quantitative attribute.

Yongzheng Z. and Xiaochun Y. (2004).
A NEW VULNERABILITY TAXONOMY BASED ON PRIVILEGE ESCALATION.
In *Proceedings of the Sixth International Conference on Enterprise Information Systems*, pages 596-600
DOI: 10.5220/0002593205960600

2 DEVELOPMENT OF NEW TAXONOMIC CHARACTERS

The basis for successful classification is appropriate taxonomic characters (Simpson, 1961; Glass and Vessey, 1995). Therefore, it is important for improving existing vulnerability taxonomies to extract new taxonomic characters.

2.1 Privilege Escalation

Through analyzing prevalent attack methods and large numbers of vulnerabilities, we detect that most vulnerabilities have the following characters: an attacker in the low user-level L usually exploits a or several vulnerabilities successfully to get a certain privilege escalation, and then, arrives at the high user-level H without authorization. Obviously, the attacker's illegal escalation from L to H seriously threatens the security of computer system.

In the whole process of exploitation, an attacker often plays a certain role of system user and owns the corresponding user privilege-set. From a visitor to a system use, finally to a system administrator, the change of an attacker's role reflects the variety in his owning system resources, namely the variety in his privileges. Therefore, based on the above practical experiences and the idea that different roles of system users have their privileges of different degree in operating system design, this paper introduces a new taxonomic character — the attribute of 'privilege-set'. Definition 2.1 and 2.2 give separately the concept of privilege, privilege-set (Pset) and privilege escalation (P-E) (Wang, 2002).

Definition 2.1 A privilege is a (x,m). Where, x is an object, m is a set of accessing modes of the subject to that object and m isn't null. Pset={(xi,mi)|(xi,mi) is a privilege, i=1~n}. We use Psubset to express any subset of Pset.

Definition 2.2 If a user 'Name' owning Pset exploits a certain vulnerability to gain a new Pset', and, $\exists$ x',m' $\neq \phi$, make (x',m') $\in$ Pset' $\wedge$ (x',m')$\notin$ Pset, then we argue that 'Name' makes a privilege escalation.

To an attacker, he exploits vulnerabilities to attack the computer system with the purpose of obtaining much more privileges. On the other hand, to a vulnerability, it is significant only if it gives an attacker more privileges.

2.2 User-Pset Relationship

As for a certain subject (user or user's process) in system, its owning permissions which authorize it to access all operable objects in system are a Pset. Hence, every subject can be regarded as a naming Pset. We can use (name,Pname) to express the correspondence of a subject (user) to a Pset. Here, 'name' means a user's name, and 'Pname' is the corresponding Pset of name. To a user 'name' in system, its default privilege-set is certain. So in the condition of the legal access, (name,Pname) is certain.

2.3 Classification of Psets

In (Longstaff, 1997), longstaff presents a taxonomy to classify all visitors of computer, and he uses Selection Decision Tree (SDT) to divide all visitor into the following five classes: Remote using a common service, Trusted system, User account, Physical access and Privileged access. In this paper, we use the above taxonomy of visitors for reference, and from the other angle, combine visitors with Psets to classify Psets of all possible users in system by user's roles. We also adopt the method of SDT to make this classification. SDT for user's role classification is given in Figure 1.

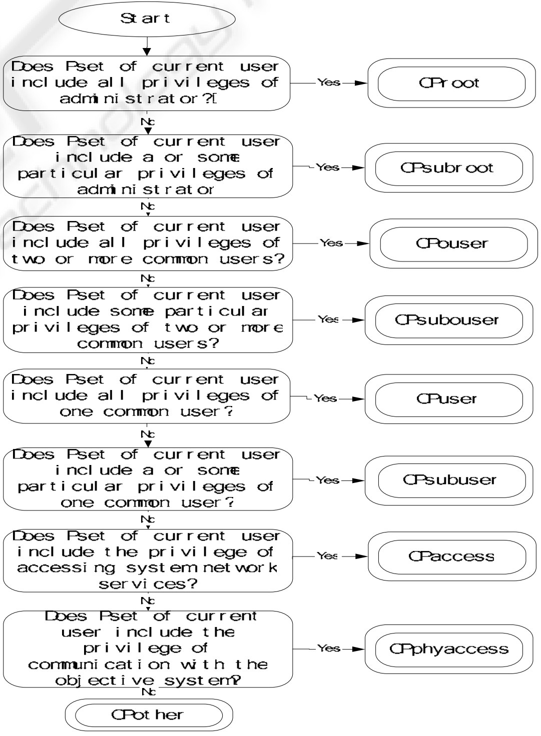
Start
Does Pset of current user include all privileges of administrator?
Yes
CProot
No
Does Pset of current user include a or some particular privileges of administrator
Yes
CPsubroot
No
Does Pset of current user include all privileges of two or more common users?
Yes
CPouser
No
Does Pset of current user include some particular privileges of two or more common users?
Yes
CPsubouser
No
Does Pset of current user include all privileges of one common user?
Yes
CPuser
No
Does Pset of current user include a or some particular privileges of one common user?
Yes
CPsubuser
No
Does Pset of current user include the privilege of accessing system network services?
Yes
CPaccess
No
Does Pset of current user include the privilege of communication with the objective system?
Yes
CPphyaccess
No
CPother

Figure 1: Selection decision tree for user's role classification

In Figure 1, the current user is a broad conception, and includes all possible users related to the objective system, such as system accounts, trusted or distrusted remote visitors, etc. Common user is any system account except system administrator. Table 1 shows the ranks and

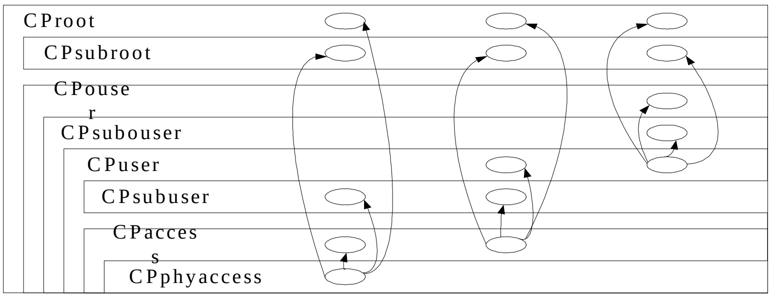

Figure 2: Relationship between Pset-Classes and P-E mode

descriptions of Psets. 'Pset-Class' is given by the classifying rule. 'Role Description' denotes user's role of Pset-Class.

There are two advantages to adopt SDT for classification: 1) To eliminate ambiguity. 2) To possess integrality. Here, we need to emphasize that 'CPother' is actually the complement of integrality. In fact, the last decision is always true. If the objective system isn't existing, or alive, or in control, or interactive with a user, then we argue that these phenomena are not significant for us. So, on the premise of significance for us, another eight classes should be able to include privileges of all roles, and 'CPother' should be null-set. This is what we anticipate.

According to the classifying rule, we can get:

CProot>CPouser>CPsubouser>CPuser>CPaccess>CPphyaccess (a)
CProot>CPsubroot (b)
CPuser>CPsubuser (c)

Where, CPb>CPa means, $\forall$ Pset$\in$CPa, must $\exists$ Pset'$\in$CPb $\wedge$ Pset'$\supset$ Pset. Figure 2 shows visually the above containing relation between Pset-Classes and P-E mode in which we give all possible and direct P-Es arising from vulnerabilities.

3 MULTIDIMENSIONAL QUANTITATIVE TAXONOMY BASED ON P-E

It is greatly difficult to describe precisely a vulnerability. The simple and effective solution to resolve this problem is to add new attributes for each vulnerability. To evaluate each attribute of vulnerabilities is actually to classify vulnerabilities by that dimensional attribute. This paper gives six quantitative attributes and ten descriptive attributes.

Table 1: Ranks of privilege-sets

Pset-Class	Role Description
CProot	System administrator. To manage all system resources, such as system devices, system files and system processes, etc.
CPsubroot	User which possesses Psubset of administrator.
CPouser	Two or more system common users.
CPsubouser	User which contains Pset of any common user and Psubset of other common users.
CPuser	Any common user which is created by administrator.
CPsubuser	User which possesses Psubset of any common user.
CPaccess	Remote visitor which may access network services, is a usually trusted visitor. It can communicate data with services and scan system.
CPphyaccess	Remote visitor which may interact with the objective system in physical layer, is a usually distrusted user or user outside firewall.

3.1 Pset Attribute and Its Quantitation

From the concept of P-E, we can see that in the process of exploiting a vulnerability, an attacker always escalates from a Pset to another one. So we design two attributes 'Cpremise' and 'Cconsequence' for our taxonomy.

'Cpremise' means the class which the premise Pset belongs to. And the premise Pset is the necessary Pset which an attacker need for attempting to exploit a vulnerability. Only when an attacker possesses the premise Pset, it is possible to exploit successfully that vulnerability.

'Cconsequence' means the class which the consequence Pset belongs to. And the consequence

Pset is the Pset which an attacker successfully exploits a vulnerability to produce. The consequence Pset reflects the harmful degree of vulnerabilities in privilege layer. It is emphasized that this Pset derives from the direct P-E, not a series of P-Es.

Based on the above two attributes, we may roughly describe the process of a series of P-Es, moreover, we can detect the correlation between vulnerabilities. The classifying rule of these two attributes sees also Figure 1.

Because of differences of important degree of user's role, the importance of its Pset is also different. We quantitate the importance of these Psets to express profoundly harmful degrees of vulnerabilities. The quantitation of Pset-Class is given in Table 2. 'Value' is decimal fraction range from 0.0 to 1.0.

How is a quantitative criterion made? We will abide by the following two principles: (1) The quantitation must satisfy (a), (b) and (c); (2) We argue that to a great extent, establishing the criterion reflects benefits of users using the vulnerability taxonomy. Different users want different criterions. Therefore, based on satisfying principle (1), we give the values which we want. Then According to feedbacks from practices, we may continually improve our quantitation.

3.2 Security Attribute and Its Quantitation

To express influences of vulnerabilities on confidentiality (C), integrality (I) and availability (A) of system security in the process of P-E, we introduce three security attributes—C, I and A into each Pset-Class. We make reference to (Wang, 2002) and make some proper modification. Table 2 gives ranks and quantitation of multidimensional impacts of vulnerabilities. We use the value increment of Cpremise and Cconsequence of each vulnerability to evaluate separately each security attribute.

Table 2: Ranks and quantitation of multidimensional impacts of vulnerabilities

Ranks	Psets-Class	Value	Confidentiality	Integrality	Availability
1	CPphyaccess	0.0	Validate: system is alive		
2	CPaccess	0.1	Validate types and versions of OS and services		In the condition of peer attacks, the load or performance of some services, processes or system will decrease.
3	CPsubuser	0.2	Read some particular files or memory of a certain common user.	Use trash to append, modify, delete,or create some files or process space of a certain common user.	To overwrite,modify or delete some files or process space of a certain common user leads to crash or unavailability.
4	CPuser	0.4	Read all files or memory of a certain common user.	Set up user-level Trojan horses;Use trash to append, modify, delete, or create all files or process space of a certain common user.	To overwrite, modify or delete all files or process space of a certain common user leads to crash or unavailability.
5	CPsubouser	0.5	Read some particular files or memory of some common users.	Use trash to append, modify, delete,or create some files or process space of some common users.	To overwrite,modify or delete some files or process space of some common users leads to crash or unavailability.

Table 2: Ranks and quantitation of multidimensional impacts of vulnerabilities (Continue)

Ranks	Psets-Class	Value	Confidentiality	Integrality	Availability
6	CPouser	0.6	Read all files or memory of some common users.	Use trash to append, modify, delete,or create all files or process space of some common users.	To overwrite,modify or delete all files or process space of some common users leads to crash or unavailability.
7	CPsubroot	0.8	Read some system files, kernel or system process space.	Use trash to append, modify, delete,or create some system files or system process space.	Make some system files, system processes or modules unavailable;system is hung up, rebooted or crashed
8	CProot	1.0	Read all files, and monitor all system activities.	Set up root-level Trojan horses; append, modify, delete,or create all files or processes.	System is unreturnably crashed.

3.3 Attack-Complexity and Its Quantitation

Influences of a vulnerability on system security are related to attack-complexity. On the premise of possibility to make a P-E, if a vulnerability is exploited easier, more attackers can exploit it to endanger system security, and so, its bad impacts on the security are higher (Wang, 2002). Therefore, we introduce the attack-complexity attribute. Ranks and quantitation of attack-complexity of vulnerabilities are given in Table 3.

Table 3: Ranks and quantitation of attack-complexity of vulnerabilities

Ranks	Value	Character Description
E7	1.0	Existing attack tools and detailed attack approaches
E6	0.8	Custom available attack tools and have detailed attack approaches
E5	0.6	No existing attack tools, but have detailed attack approaches
E4	0.5	Open report, and describe roughly attack methods
E3	0.2	Open report, and mention possible attack methods
E2	0.1	Open report, but no attack methods
E1	0.05	Open report, but the attack only exists in theory; no vulnerabilities reported

4 FUTURE WORK

Although we have acquired some achievements, we have still plenty of work to do. Our future work will focus on the following: (1) In order to support the security evaluation for computer system better, we need more detailed description of privileges and attack modes. (2) We will do many tests or research new methods to validate the scientificity of quantitative criterion of vulnerabilities. (3) Based on this taxonomy, we will establish an effective security evaluation model for computer network and system.

5 CONCLUSION

This paper integrates a privilege-escalating based vulnerability taxonomy with multidimensional quantitative attribute. This taxonomy has a particular effect on the quantitative support on various sides, the detection of correlation between vulnerabilities, and the attack-complexity of vulnerabilities. It is greatly significant to make further security evaluation and other security practices.

REFERENCES

Aslam, T., 1995. A Taxonomy of Security Faults in the Unix Operating System. M.S.thesis, Purdue University.

Aslam, T., Krsul, I., Spafford, E.H., 1996. Use of A Taxonomy of Security Faults, *the 19th National Information System Security Conference*, Baltimore, Maryland, October, 22-25.

Bishop, M.D., Bailey, D., 1996. A Critical Analysis of Vulnerability Taxonomies. Tech. Rep. CSE-96-11. Department of Computer Science at the University of California at Davis. September.

Knight, E., Hartley, B.V., 2000. Is Your Network Inviting an Attack? Internet Security Advisor. May/June: 2-5.

Jiwnani, K., Zelkowitz, M., 2002. Maintaining Software with a Security Perspective, *Proceedings of the International Conference on Software Maintenance(ICSM'02)*, pp. 194-203.

Wang, L. D., 2002. Quantitative Security Risk Assessment Method for Computer System and Network. Ph.D. thesis, Harbin Institute of Technology.

Simpson, G. G., 1961. *Principles of Animal Taxonomy*. Columbia University Press.

Glass, R. L., Vessey, I., 1995. Contemporary Application-Domain Taxonomies. *IEEE Software* 12, 4 (July), 63-76.

Longstaff, T., 1997. Update: CERT/CC Vulnerability Knowledgebase. Technical pre-sentation at a DARPA workshop in Savannah, Georgia.