Patching Power System Software Vulnerability Using CNNVD

Qin-qin WU¹, Li-hao WEI¹, Zhi-qiang LIANG¹, Zhi-wen YU¹, Min CHEN¹, Zhi-hua CHEN²,* and Jin-jun TAN²

¹Information Center, Guangdong Power Grid Company Limited, Guangzhou, China
²Guangdong Information Technology Security Evaluation Center, Guangzhou, China

*Corresponding author

Keywords: Power system, Software vulnerability, CNNVD, Vulnerability patching.

Abstract. Both the attractiveness of power systems as targets of cyberattack and their vulnerability to remote attack via digital networks are evident from recent world events. Vulnerabilities in software represent a serious risk for power systems. The number of vulnerabilities is increasing rapidly due to the development of new hacking techniques. In order to reduce threat impact of software vulnerability, companies should use vulnerability management frameworks. Software vulnerability patching is one of the key activities of vulnerability management. This paper proposes a novel software patching mechanism based on CNNVD for power systems. This mechanism can patch power systems software vulnerability quickly.

Introduction

Software vulnerabilities manifest in many forms, including race conditions, buffer overflows, integer overflows, dangling pointers, poor input validation (e.g., SQL injection, cross-site scripting), information leakage, violation of least privilege and other access control errors, use of weak random numbers in cryptography, protocol errors, and insufficient authentication[1-2]. Discovering vulnerabilities has become an extremely widespread activity, new threats are constantly being faced. So companies are forced to manage risks by applying various approaches. Since each vulnerability presents different level of threat, it is necessary to apply an appropriate prioritization policy. The National Infrastructure Advisory Council (NIAC) introduced an open standard Common Vulnerability Scoring System (CVSS) in 2005, which allows quantitative assessment of vulnerabilities’ severity. CVSS is widely accepted today and has become an integral part of automated vulnerability management tools based on SCAP protocol2.

Vulnerabilities in software represent a serious risk for information systems. According to the National Vulnerability Database (NVD), which maintains records of all acknowledged vulnerabilities of software products on the market, more than 90,000 vulnerabilities have been discovered since 1997. The number of vulnerabilities is increasing rapidly due to the development of new hacking techniques. Between 2010 and 2015, around 80,000 vulnerabilities were newly registered in the major database known as the CVE (Common Vulnerability Enumeration) [3]. In recent years, the scope of security threats has also been expanded. As cataloged in the National Vulnerability Database, prominent recent examples of serious software vulnerabilities include Heartbleed (CVE-2014-016), Shellshock (CVE-2014-6271), the glibc buffer overflow (CVE-2015-7547), VENOM (CVE-2015-3456), and Microsoft Malware Protection Engine vulnerability (CVE-2017-0290). According to the website CVE Details, there have been 937 vulnerabilities between January and August 2017 to date. Ten percent of those vulnerabilities were in the “critical” (highest severity) category with assigned Common Vulnerability Scoring System (CVSS) scores of 9 or 10.

Both the attractiveness of power systems as targets of cyberattack and their vulnerability to remote attack via digital networks are evident from recent world events. While policy makers seek means to deter such attacks politically, an effective way to reduce their attractiveness as targets is to patch their vulnerability as soon as quickly. This can be done; we present a mechanism based on CNNVD (China National Vulnerability Database of Information Security, CNNVD)[4], this mechanism can patch power systems software vulnerability quickly.
The rest of the paper is organized as follows. Section 2 is the review of recently related research. Section 3 is a brief introduction of CNNVD. The proposed mechanism is introduced in section 4. In section 5, experiment results will be discussed with details.

Related Research

Software vulnerability is a kind of flaw that arises in software or is a hole in the security of the software that allows an attacker to exploit that flaw. Unlike bugs, software vulnerability can affect a whole network thereby allowing unauthorised access to the database itself. In order to reduce threat impact of software vulnerability, organisations should use vulnerability management frameworks. Vulnerability management involves the cyclical practice of identification, classification, remediation, and mitigation of vulnerabilities. Repetition of this process helps in mitigating the vulnerabilities in the software effectively. A risk can lead to significant loss as that of a vulnerability but it is not mandatory that all vulnerabilities will involve a risk. There can be vulnerabilities without risk especially when the affected asset has no value. An exploit exists for an exploitable vulnerability with one or more instances of fully implemented attacks. An exploit exists for an exploitable vulnerability. An exploit is a code that an attacker creates to target a software vulnerability in applications like multimedia, security programs. There is also a window of vulnerability that decides a time between when a security flaw is introduced in the system that compromises system security and the time when an attacker is disabled. Apart from software vulnerabilities, vulnerabilities can also exist in hardware, site or personnel.

Software vulnerability patching is one of the key activities of vulnerability management, and only informed resource allocation decisions will yield effective defenses against threats. In order to rate the impact severity of specific software vulnerabilities, scoring systems such as CVSS[5], has been developed, depending on their characteristics. In [6], the authors investigate the use of game theory as an alternative to CVSS-only vulnerability patching strategies. The initially studied game is a two-target security game, also known as a search game in the literature. The game theoretic analysis is used as a filter to narrow down the number of vulnerabilities that need to be considered for patching, e.g. using traditional vulnerability scoring techniques such as CVSS. In [7], the authors focus on the attacker’s perspective inspired by game theory, using a slightly different game, but in addition they capture concurrent and simultaneous attacks and also model return of investment equations. In [8], the authors present a strategy for vulnerability patching based on sound theoretical foundations, in particular when using CVSS aided by game theory.

In this paper, we design a novel software patching mechanism based on CNNVD for power systems, which can automatic update of vulnerability information. Our mechanism can patch power systems software vulnerability quickly.

CNNVD

The China National Vulnerability Database of Information Security (CNNVD) was officially established on October 18, 2009. In order to effectively perform the functions of vulnerability analysis and risk assessment, CNNVD is responsible for the construction of the national information security vulnerability database for operation and maintenance. To provide flexible and diverse information security data services to the state, industry and the public, and to provide basic services for Chinese information security. CNNVD is the national information security vulnerability data management platform for Chinese information security assessment center, which is responsible for the construction and maintenance of national information security vulnerability data management platform, with the support of special state funds, in order to effectively perform the functions of vulnerability analysis and risk assessment. CNNVD, by means of self-mining, social submission, collaborative sharing, network collection and technology detection, combines social forces such as government departments, industry users, security vendors, universities and scientific research institutions to develop information security vulnerabilities in software and hardware.
systems involving mainstream applications, operating systems and network devices at home and abroad. Collection, analysis and verification, early warning and notification, repair and elimination control work, and the establishment of a standardized vulnerability assessment and disposal process, smooth information sharing and notification mechanism, as well as a perfect technical collaboration system. The vulnerability disposal involves thousands of major manufacturers at home and abroad, covering the government, finance, transportation, industrial control, health care and other industries. Industry has provided important technical support and data support for security assurance of key industries and infrastructure in China. It has played an important role in enhancing the industry-wide information security analysis and early warning capabilities, and improving the network and information security assurance in China.

**Patching Power System Software Vulnerability Mechanism**

Business flow of our mechanism is shown in figure 1. The vulnerability information management module has a common interface to support the connection with third-party vulnerability information platforms to obtain the latest vulnerabilities. The module has an automated inspection module that crawls the vulnerability information published by the CNNVD, according to the set vulnerability crawling rules. The module also supports batch import or manual entry to add vulnerability information.

![Figure 1. Business flow of mechanism.](image)

Sources of information assets include: 1) Interfacing with existing asset information management related platforms (such as ITSM systems, etc.) to obtain asset-based data. 2) Dispatching the distributed asset collection radar, obtaining asset attributes and fingerprint information, and monitoring asset changes. 3) Support batch import or manual entry to add asset information.

Sources of threats include: 1) Establishing vulnerability scanning tasks on the platform, conducting vulnerability scanning, and exploiting vulnerability. Vulnerability scanning supports different granularity vulnerability scans, including: full vulnerability library scanning, scanning by vulnerability category, specifying scans for specific vulnerabilities. And more. 2) When the vulnerability database or asset library changes, obtain the information in the internal network of the
information through asset matching and rapid verification of the vulnerabilities in the network, and automatic measures threat levels, and form internal threat intelligence.

Visualization of internal threat intelligence is mainly based on map visualization, visualization based on different latitudes, visualization of vulnerability response based on different time points, recurring visualization based on the same type of vulnerability, visualization based on vulnerability validation, visualization based on the scope of the vulnerability.

Vulnerability information management defines content management for vulnerability database information, which includes attributes such as CNNVD number, vulnerability type, vendor, component, vulnerability summary, vulnerability detail, vulnerability scope, vulnerability exploit tool, POC platform type, vulnerability POC, vulnerability level, temporary solution, official solution, protection plan, external reference, CVSS 2 score, CVSS score, etc.

Vulnerability governance includes vulnerability discovery management, vulnerability threat management, vulnerability handling task management, risk level rule determination management. Vulnerability discovery management defines information about the vulnerability discovery task, which is discovered by calling vulnerability radar. It includes attributes such as risk number, target, threat name, threat type, vulnerability number, CVE number, asset name, asset owner, equipment type, vulnerability description, repair suggestion, threat level, discovery time, etc. Vulnerability handling task management generates a vulnerability handling task ticket based on the vulnerability threat information, and the work order can automatically complete closed loop processing or push to the tone on the platform. Risk level rule determination management defines risk level rule judgments, used to specify risk level when creating vulnerability handling tasks.

Conclusion
In this paper, we have designed novel software vulnerability patching strategies using CNNVD to decide whether to apply a specific patch or not. In addition, our approach has been implemented as part of an existing tool. Our first experiments with this tool and its new functionality are encouraging and there is great scope for additional investigation and further results. In our future work, we will perform our mechanism in real power system. Our research will lead to the creation of automated security assessment tools with higher and more accurate prediction of security attacks.

Acknowledgments
This work was financially supported by science and technology project which project number was GDKJXM20162130 of Guangdong Power Grid Company Limited.

References

