CERIAS Tech Report 2002-34
Running the free vulnerability notification system Cassandra
by Pascal C. Meunier and Eugene H. Spafford
Center for Education and Research
Information Assurance and Security
Purdue University, West Lafayette, IN 47907-2086

CERIAS Tech Report 2002-34

RUNNING THE FREE VULNERABILITY NOTIFICATION SYSTEM CASSANDRA

by Pascal C. Meunier and Eugene H. Spafford

Center for Education and Research in
Information Assurance and Security,
Purdue University, West Lafayette, IN 47907

Running the free vulnerability notification system Cassandra

Pascal C. Meunier and Eugene H. Spafford

Center for Education and Research in Information Assurance and Security (CERIAS)
Purdue University
W. Lafayette, IN 47907-1315

Summary

The public part of the vulnerability management cycle — publication-notification-patch — is of interest to system administrators. We describe the architecture of the vulnerability notification Cassandra system[1]. The timeliness of Cassandra notifications was evaluated by using the publication dates of CERT Incident Notes as approximations for the dates when vulnerabilities are widely exploited. We found that notifications sent by Cassandra in 2001 (until November) provided a forewarning of 60 days on average. However, these notifications were not always timely. An analysis of the vulnerability information flow identified sources of undesirable delays. A new Cassandra service, CVE Change Logs, was created to report daily changes to the CVE and bypass some sources of delays. Other efforts by MITRE mitigated other sources of delays and consolidated changes on their web site.

An unexpected finding of this study is that the timing and the number of vulnerabilities involved in the method of disclosing vulnerabilities can contribute to notification delays caused by the limited processing capacity of intermediates and the finite patching capability of system administrators. We conclude that large batch processing of vulnerabilities contributes to notification and patching delays and is undesirable. For the same reasons, randomly timed disclosures of vulnerabilities should be undesirable, suggesting the creation of a focused mechanism for the disclosure of vulnerabilities.

[1] https://cassandra.cerias.purdue.edu

1. Introduction

Many systems remain vulnerable to security flaws months or even years after corrections become available; this, and not public disclosure of vulnerabilities, is the primary driving force for widespread intrusions [Arb00]. Vulnerability and exploit life cycles of 2 to 3 years are observed in DoS (Denial-of-Service) attacks [CER01]. One reason for this is that keeping up-to-date is a time-consuming task. Automated patch systems are in development (e.g., the Lawrence Livermore's Secure Software Distribution System) or have been proposed [Car98,Liu00]. However, they are not entirely trusted by system administrators concerned that the patch might cause other problems. Moreover, third-party-controlled remote patching systems may have vulnerabilities of their own. As an alternative, vulnerability notification services are available but most are too expensive for many users (e.g., academic and home users).

The Cassandra service is the first tool to provide free, customized vulnerability notifications, based on NIST's ICAT metabase. It allows users to create saved profiles of services and applications. Cassandra can then notify users by e-mail of new vulnerabilities relevant to those profiles. Queries can also be performed live through SSL. A special kind of query, "incremental query," shows only the new results since the last time the query was run. However, these results may be missing recently-discovered vulnerabilities not yet available from ICAT, and will be missing vulnerabilities that have not been made public. Vulnerabilities are included in ICAT as a result of additions to MITRE's CVE (Common Vulnerabilities and Exposures) [Man99, Mar02]. Therefore, the custody chain of vulnerabilities, disclosure-CVE-ICAT-Cassandra, introduces a delay in the notifications. The purpose of this paper is to investigate how significant this delay is, and how useful are the notifications provided by Cassandra. Because no information was available to us prior to disclosure, we focused on the public part of the vulnerability management cycle, publication-notification-patch and the exploitations that happen in the meantime.

2. Models and Systems

2.1. Cassandra architecture

The Cassandra system searches the contents of a local copy of the ICAT database for product names or keywords; the combination of products and keywords is stored in a profile. Searches may be live or automated; live searches are secured through the SSL protocol. Automated searches generate e-mails when new vulnerabilities have been found. The contents of the e-mails are selectable to be a brief note stating that new profiles were found, or a detailed list of the

vulnerabilities found. As of March 3, 2002, 48% of profile submissions were requests for e-mail notifications; the other 52% were live queries, were experimental or were unused. Of the profiles with e-mail notifications, 10% requested only a brief notice and 90% requested the list of vulnerabilities found. Cassandra is based on PHP, Apache, MySQL and may run on any Unix-type system (currently Linux).

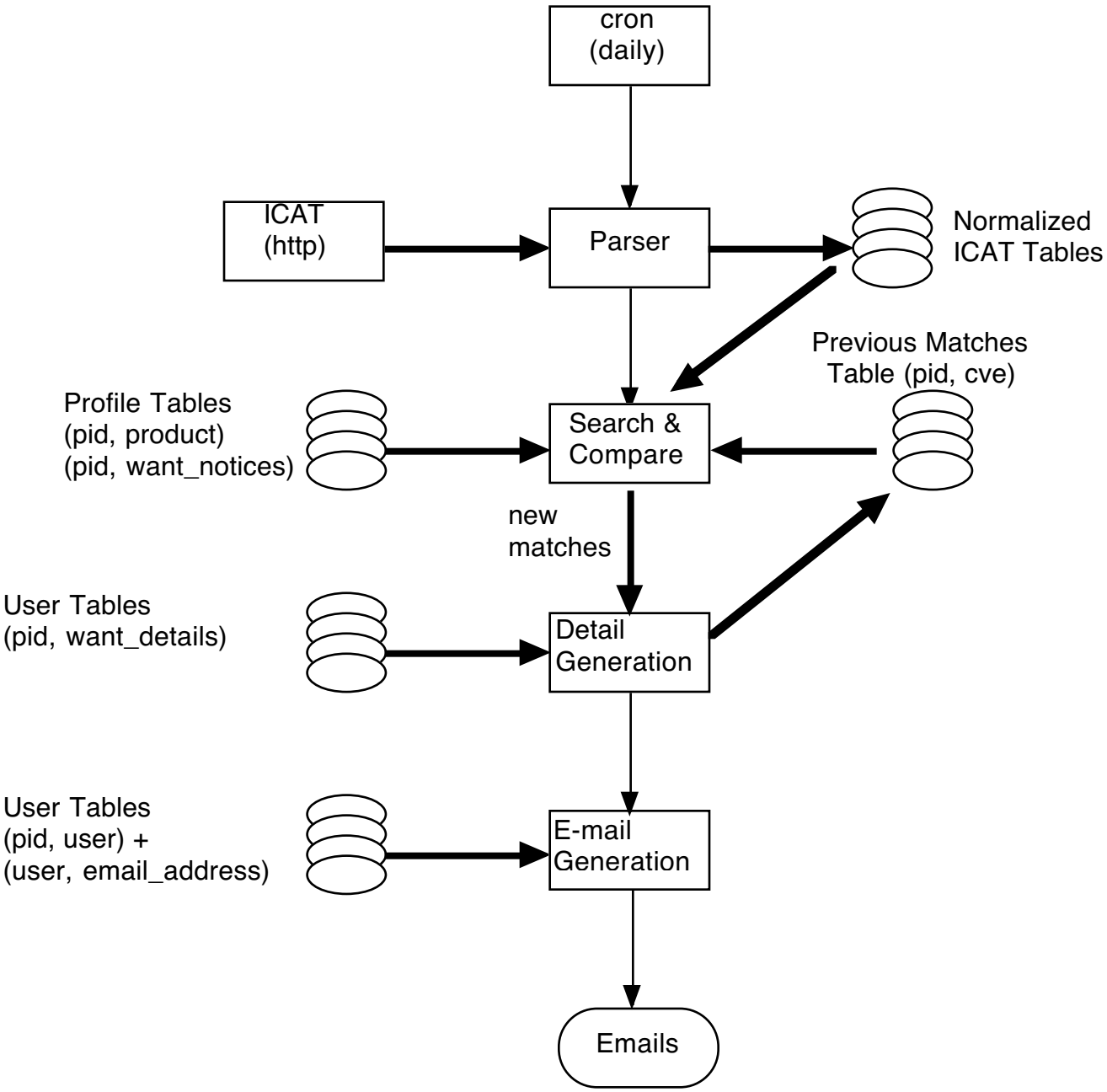

Figure 1. Architecture of the Cassandra system. "pid" refers to a unique profile identifier.

The Cassandra system imports the ICAT database daily, parses it and stores the data in normalized tables. Profiles that have the e-mail notification flag set are then run against the database. New matches are recorded in the database and in a local variable for the generation of the appropriate e-mail message. If there were any new matches, the relevant e-mail addresses are obtained and the e-mail is sent.

2.2. Operational Policies of the Cassandra system

2.2.1. Account Creation

Cassandra accounts can be created at will and may be anonymized through the use of throwaway e-mail addresses and aliases. However, e-mail addresses are authenticated by sending a unique code in an e-mail that the user copies and pastes into a special field of the login system (the "challenge" password). This discourages subscribing unwilling third parties as pranks or by malice. There may be only one account per e-mail address to prevent duplicate accounts, but there may be multiple profiles per account.

2.2.2. Account Termination

Accounts that bounce e-mails back to the Cassandra system with a permanent error message (not a temporary delivery failure) get deleted. At the beginning of 2001 we tried to prune inactive accounts, but this was unpopular as our definition of "inactive" did not match actual use. As of March 3, 2002, people who owned profiles with e-mail notifications had not logged in on average for almost 7 months (206 days). Assuming that having active e-mail notifications implies that the accounts have not been abandoned, and that profiles are still in use and valid, this implies that profiles remain unchanged for at least that long; this was a surprise to us. This could be caused, in part, by the absence of version numbers in the profiles. In conclusion, the requirement that users log in at least once every three months to maintain an account was thought excessive and was rescinded.

2.2.3. Mailing Lists

Some users actually give a mailing list address as their e-mail address for notifications. This makes it more difficult to estimate the number of users of the Cassandra system, but is within the bounds of our acceptable use policy. Mailing lists have caused difficulties with the account termination policy when the e-mail address of a list member became invalid and e-mails were bounced back to the Cassandra system. However, such cases are detected by comparing the e-mail address used to send the original e-mail and the e-mail address that bounced,

which are different. However, cases of aliases, mail forwarding, and other cases where there are non-matching addresses make this process labor-intensive. Manual methods will not scale well should the subscriber base grow significantly.

2.2.4 Confidentiality and Privacy

CERIAS will not reveal profiles to any third party, unless obliged to do so by law. CERIAS personnel are not allowed to view individual profiles except as an unavoidable side-effect of performing system maintenance. Login failures are logged with the IP address of the attempt and we reserve the right to deny access from addresses with too many login failures. The system uses a web cookie authentication mechanism that logs the IP addresses that received the cookies. Profiles are run against a local database updated from ICAT and are never sent outside the system. User connections to Cassandra are performed over encrypted SSL links, which protects the cookies.

We have attempted to provide a protected system to keep user profiles from external view or alteration. We realize, however, that there are ways to view profile data that are not under our control (e.g., hacking into a client's system). This is balanced with the knowledge that a malfeasor with access to a target system's information could obtain the same overall vulnerability data by searching the ICAT database. Thus, we do not believe that the data held in our system poses any significant new threat to end users that would warrant more involved security mechanisms.

2.2.5. Timeliness

Results may be missing recently discovered vulnerabilities that are not yet available from ICAT, and will be missing vulnerabilities that have not been made public. Because the contents are derived from NIST's ICAT servers, CERIAS cannot make any representation other than a best effort delivery of the contents available from ICAT. With these limitations in mind, and considering that this is a research effort, we have disclaimed any liability for missing or inaccurate information.

2.3. Model of the vulnerability information flow before Nov. 2001

The CVE was originally designed to employ a two-step validation mechanism for vulnerability records [Man99]. Candidates are vulnerability records that require validation by the board of editors, in the form of multilevel voting (Abstain, Accept, Modify, etc.). When a candidate (with the "CAN" prefix) passes the qualification process, it becomes a full CVE entry (with "CVE" as its prefix).

The generation and the publication of candidates is often time-consuming because of the effort required for analysis of the vulnerabilities and because of the deliberations involved in the CVE content decision process. One source of information for the generation of candidates is existing databases whose owners are cooperating with the CVE effort to produce "legacy" candidates. Public sources of information are also harvested for vulnerability information, but this requires sorting the information for relevance and accuracy. Finally, MITRE developed a process by which candidate numbers may be reserved in advance by discoverers of vulnerabilities.

The publication of new CVE candidates is accomplished in two different ways:

- MITRE proposes candidates to the board of editors at biweekly or (most often) monthly intervals, so that the candidates can be grouped by clusters with similar characteristics.
- New candidates may be published as references in advisories by using the candidate reservation process.

The list of candidates can be downloaded from the CVE web site[2].

This process divides the CAN state into five sub-states: reserved, non-reserved, RP (reserved and published by a third party, but the record has not been updated yet), WABNYP (web-accessible but not yet proposed), and proposed (to the board). Before November 2001, non-reserved candidates were not available on the CVE web site until they had been proposed to the board. Candidates from CNAs become "WABNYP" upon publication of their advisories.

The sources of delays we identified in sending our notifications were:

a. The generation of candidates. Legacy candidates are generally of low interest to subscribers of the Cassandra service; however, whenever MITRE's efforts have been focused on old vulnerabilities, the processing of new vulnerability information from public sources has been less timely[3].

b. The complicated manner in which new candidates were published. ICAT simplified the problem by obtaining its vulnerability information from the list of candidates proposed to the board, and therefore ICAT added vulnerability information in blocks, resulting in approximately monthly updates to ICAT. However, ICAT updated its database after all records had been processed instead of providing a flow, resulting in further delays. It follows that the Cassandra notifications also happened in blocks after some

[2] http://cve.mitre.org

[3] Christey, S., personal communication

additional delay for processing.

c. Limited cooperation with the CVE effort by vendors and vulnerability publishers. MITRE added the capability to use confidential information to pre-generate candidates before the publication of advisories. In this manner, vendors and major advisory releasers can include candidate numbers in advisories. On the CVE web site, those candidates are marked as "** RESERVED **" in their description, until the advisories are published. By using confidential channels, the latency between publication and the generation of a CVE entry can be pre-empted. Without the cooperation from vendors and vulnerability publishers, the CVE content team is using information that is already delayed, with the substantial handicap of having to poll information sources.

   To make the use of candidate reservation more attractive, MITRE also has the capability to designate Candidate Numbering Authorities (aka CNAs) that are given an "empty" block of candidates and can assign them to issues as they see fit. Assuming that CNAs would produce candidates for inclusion in their advisories, this would also pre-empt latency between publication and the generation of a candidate. However, duplications between CNAs are possible, unless the duties of the various CNAs do not overlap or there is a diligent coordination across all parties who are involved in the disclosure process. The CNAs also have to be trained to make the same abstraction choices and content decisions as MITRE's content team does.

d. Technical problems, such as the ICAT web site being inaccessible, the wrong version of the database being available for download, or failures in the Cassandra system. These problems were mitigated by having the results of every script being e-mailed to the operator of the Cassandra service, and by diligent attention most problems were solved within 24 hours of their detection; the ICAT team was also highly responsive and efficient when contacted.

2.4. Measurements of timeliness

We used the publication dates of CERT Incident Notes as approximations for the dates when vulnerabilities are widely exploited. This is only an approximation because only incidents that are discovered can be reported; because only a fraction of the incidents are reported to CERT (perhaps inconsistently); because there could be jitter between exploits reaching a given importance and the publication date of the advisory; and because the mechanism for the publication

by CERT of incident notes might not be consistent. Moreover, CERT publishes only a dozen or so incident notes every year, and not every one relates to specific vulnerabilities, so our data set is limited. However, the publication of these notes is a strong signal that matching vulnerabilities (when present) should be patched as soon as possible because of their impact.

CERT also publishes advisories announcing the presence of vulnerabilities, and these are more numerous than Incident Notes. For consistency with the use of CERT incident notes, we used the dates of CERT advisories to measure the improvements provided by the Cassandra service. Because the timeliness of those advisories is not guaranteed, we also used the first publication we could find as an alternate measurement. Vulnerability records released in 2001 were selected from the CVE (and candidates) by searching for references to CERT advisories.

3. Results

3.1. Forewarning provided by Cassandra notifications before November 2001

Table 1 shows the publication dates of CERT incident notes compared to the dates of notifications by Cassandra. The average forewarning was 60 days. Whereas this appears to be a significant amount of time, it is only an average. For instance, the Cassandra notifications for "Carko" and the first version of "Code Red" were late[4] .

For the sake of being able to measure the improvements provided by the corrective measures described in 3.2, we determined the delay between sources of vulnerability information and the notifications by the Cassandra system up to November 2001 (Table 2). The delay was either 40 or 46 days depending on whether CERT advisories or the earliest publication was used.

[4] CANs were quickly available for the vulnerabilities exploited by Carko and Code Red but were "WABNYP" for a long time (Christey S., personal communication). The CVE Change Log (3.2.2) would therefore have prevented Cassandra notifications from being late.

CVE	Exploit	Exploit Date	Cassandra Date	Warning (days)	Notes
2001-0010	IN-2001-03	2001-03-30	2001-02-13	45	'1i0n' worm &
2001-0011	IN-2001-03	2001-03-30	2001-02-13	45	'cheese' worm
2001-0012	IN-2001-03	2001-03-30	2001-02-13	45	
2001-0013	IN-2001-03	2001-03-30	2001-02-13	45	
2001-0144	IN-2001-12	2001-11-05	2001-03-13	237	
2001-0236	IN-2001-04	2001-04-24	2001-05-05	-11	Carko
2001-0333	CA-2001-26	2001-09-18	2001-06-28	82	Nimda
2001-0500	IN-2001-08	2001-07-19	2001-07-21	-2	Code Red
			Average	60.75	

Table 1. Time interval between notifications by Cassandra and exploit dates in the year 2001. “CVE” refers to the number given to a vulnerability in MITRE’s Common Vulnerabilities and Exposures.

CVE	CERT Advisory	CERT Date	Cassandra Date	Delay 1 (days)	Earliest Date	Delay 2 (days)
2000-0889	CA-2000-19	2000-10-25	2001-02-13	111	2000-10-24	112
2000-1039	CA-2000-21	2000-11-30	2001-01-11	42	2000-11-30	42
2001-0008	CA-2001-01	2001-01-10	2001-02-13	34	2001-01-10	34
2001-0010	CA-2001-02	2001-01-29	2001-02-13	15	2001-01-29	15
2001-0011	CA-2001-02	2001-01-29	2001-02-13	15	2001-01-29	15
2001-0012	CA-2001-02	2001-01-29	2001-02-13	15	2001-01-29	15
2001-0013	CA-2001-02	2001-01-29	2001-02-13	15	2001-01-29	15
2001-0236	CA-2001-05	2001-03-30	2001-05-05	36	2001-03-14	52
2001-0241	CA-2001-10	2001-05-02	2001-06-28	57	2001-05-01	58
2001-0247	CA-2001-07	2001-04-10	2001-06-19	70	2001-04-09	71
2001-0248	CA-2001-07	2001-04-10	2001-06-19	70	2001-04-09	71
2001-0249	CA-2001-07	2001-04-10	2001-06-19	70	2001-04-09	71
2001-0328	CA-2001-09	2001-05-01	2001-06-28	58	N/A	58
2001-0333	CA-2001-12	2001-05-15	2001-06-28	40	2001-05-15	40
2001-0353	CA-2001-15	2001-06-29	2001-07-21	22	2001-06-19	32
2001-0500	CA-2001-13	2001-06-19	2001-07-21	32	2001-06-18	33
2001-0537	CA-2001-14	2001-06-28	2001-07-21	23	2001-06-27	24
2001-0552	CA-2001-24	2001-08-15	2001-09-22	38	2001-06-08	106
2001-0554	CA-2000-21	2001-07-24	2001-08-18	25	2001-07-18	31
2001-0718	CA-2001-28	2001-10-08	2001-10-31	23	2001-10-04	27
			Average	40.5		46.1

Table 2, Time interval between the publication of vulnerability information and notification by Cassandra in the year 2001, broken down by individual vulnerability. “Delay 1” and “Delay 2” respectively compare CERT advisories (CA) and the earliest publication we found related to the Cassandra notifications. “CVE” refers to the number given to a vulnerability in MITRE’s Common Vulnerabilities and Exposures. Only records listing a matching CA were used.

3.2. Changes to the vulnerability information flow in November 2001

The analysis of the information flow carried out in November 2001 resulted in several changes.

3.2.1 Changes at MITRE

In November 2001 and independently from this work, MITRE dedicated staff to monitoring current public sources and creating candidates from them, using a more efficient process. It is expected that the latency for the generation of CVE candidates will be reduced to 1 to 2 weeks[5] . MITRE also made non-reserved candidates available on the web site on the day of their number assignment. These changes mitigate the sources of delays 2.3 (a) and part of 2.3 (b). Reserved candidates will continue to be updated on the web site within a day or two of their publication (assuming that the discloser notifies MITRE).

3.2.2. The CVE Change Log, a new Cassandra service

Another change is the creation of a CVE update mailing list and web service in Cassandra. This new service provides a daily list of changes in the CVE database. Among the subscribers is ICAT. This removes delays of up to one month introduced by the assembly of vulnerabilities into clusters for their proposal to the board of editors, and mitigates delays in 2.3 (b).

3.3. Forewarning provided by Cassandra notifications after November 2001

For public sources of information (e.g., BugTraq), in theory, the delay should be 10 days for MITRE to update the CVE, 1 day for the CVE-update mailing list, 3 days for ICAT and 1 day for Cassandra, for a total of 15 days . For reserved candidates, the delay should be 1 day for the CVE, 1 day for the CVE-update mailing list, 3 days for ICAT and 1 day for Cassandra, for a total of 6 days.

After the implementation of the scripts for the CVE-update mailing list and the CVE Change Logs web site, two bugs were found. These bugs decreased NIST's confidence and reliance on the service, which was not used by NIST until March 2002. As a result, some 15 vulnerabilities in the January 2002 CVE Change Logs appeared in ICAT about a month later than they could have, as shown in Table 3. Conversely, this indicates that using the CVE Change Logs feature would have decreased delays by about a month, and that it is therefore a worthwhile improvement. This could decrease the delays measured in Table 2 from 46 days to 18 (46 -31 +3), close to the theoretical delay of 15 days. For reference, there were 200 vulnerabilities added to ICAT in January 2002, and 178 in February

[5] Christey, S., personal communication

2002, and an average of 128 vulnerabilities per month added to ICAT in 2001.

CVE #	Change Logs	ICAT download	Delay (days)
2002-0001	2002-01-03	2002-02-14	42
2002-0002	2002-01-03	2002-02-14	42
2002-0005	2002-01-08	2002-02-14	37
2002-0007	2002-01-10	2002-02-14	35
2002-0008	2002-01-10	2002-02-14	35
2002-0009	2002-01-10	2002-02-14	35
2002-0010	2002-01-10	2002-02-14	35
2002-0011	2002-01-10	2002-02-14	35
2001-0891	2002-01-18	2002-02-14	27
2002-0038	2002-01-18	2002-02-14	27
2002-0043	2002-01-23	2002-02-14	22
2002-0044	2002-01-23	2002-02-14	22
2002-0045	2002-01-23	2002-02-14	22
2002-0046	2002-01-23	2002-02-14	22
2002-0047	2002-01-23	2002-02-14	22
		Average	31

Table 3. Comparison of when vulnerabilities where noted in the Change Logs vs. when they appeared in ICAT.

3.4. Dangerous effects of the batch processing of vulnerabilities

A large batch of 192 vulnerabilities was released by MITRE on February 2. As of March 6, 2002, 48 of these were still not in ICAT (Table 4). We suspect this is not caused by any limitation of ICAT's capacity for processing vulnerabilities because the ICAT team was able to process 370 vulnerabilities in October 2001. It is possible to conclude that whereas in January only 8% of vulnerabilities would have benefited from the CVE Change Logs, this figure is closer to 25% for February, using a conservative estimate of counting only the missing vulnerabilities.

CVE #	Change Logs	ICAT download	Delay (days)
Batch (192) 48	2002-02-02	lost/queued (?)	
Batch (192) 45	2002-02-02	2002-02-14	12
Batch (192) 99	2002-02-02	2002-02-28	26

Table 4. Processing of a batch of 192 vulnerabilities from MITRE by ICAT.

However, another interpretation is that the capacity for processing vulnerabilities by the ICAT team has a limit, which may be lower or higher at certain times as the organization undergoes changes. If R is the maximum rate at which vulnerabilities can be processed by ICAT, then the time required to process a batch of N vulnerabilities is N/R. On March 5, 2002, NIST committed to a capacity to process vulnerabilities:

> "NIST is planning to update ICAT weekly based upon recent CVE and candidate changes with a weekly limit of analyzing 40 vulnerabilities."[6]

This translates to a capacity for processing 173 vulnerabilities/month, well above the average of 128 vulnerabilities/month for 2001. However, at 40 vulnerabilities/week, processing 192 vulnerabilities would take about 34 days. This would increase the window of vulnerability of computer systems depending on Cassandra by an equivalent amount. In other word, attackers would have 34 more days in which to attack systems vulnerable to the last 40 vulnerabilities. We conclude that the large batch processing and release of CVE candidates contributes to notification delays and is undesirable.

4. Discussion and Conclusions

The window of vulnerability after a new vulnerability is publicly known is a race between attackers or developers of hostile code, and system administrators and analysts. The primary driving force for widespread intrusions is the latency between the disclosure of vulnerabilities and the application of corrections to systems [Arb00]. The participation of more vendors and advisory-releasing organizations in the CVE effort through the use of reserved candidates or alternate CNAs should provide even smaller latencies. The latency in the custody chain of vulnerabilities could be reduced from 40-46 days to about 15 on average, and as little as 6 days if reserved candidates were used (a 7-fold improvement in performance).[7]

An unexpected finding of this study is that the timing and the number of vulnerabilities involved in the method of disclosing vulnerabilities can be a considerable factor. Contributions to the windows of vulnerabilities in excess of an additional month are possible while using batch processing of vulnerabilities. Moreover, administrators also have a limited capacity to apply patches, so for a

[6] Peter Mell, personal communication, March 5, 2002.

[7] Of course, publication of vulnerabilities in mailing lists or newsgroups may result in some admins receiving notification within hours of first publication. Our focus is on the more structured notification that can be associated with ICAT and CVE.

batch of N patches or workarounds, one can expect system administrators to take N/P days to apply all of them (where P is the rate of internal testing and patching by administrators, and dependent on site, systems, vulnerability type, etc.). We suggest that vulnerabilities should be disclosed and processed in as close as possible to a continuous stream from MITRE to Cassandra (or any similar service) instead of in large batches. A further improvement might be possible if the vulnerability disclosure process was controlled instead of random. Randomly timed vulnerability disclosures might follow a distribution that could generate a large number of disclosures on the same day or in the same week[8] . As a result of the limited processing rate of MITRE's CVE process, ICAT's team, and system administrators, the window of vulnerability can be extended by this random effect. We conclude that it would be desirable (although not likely for a variety of social and political reasons) for some central authority to control the timing of vulnerability disclosures. Although such a dampening process could be adopted by MITRE or some other centralized authority, it would then compete with open publication in mailing lists and newsgroups; should there be significant lack of synchrony between the two, participation in the CVE effort could suffer.

The potential for timely notifications with ICAT and Cassandra is present but will require some more work and adjustments. The use of CVE reserved candidates by both individuals and companies will help decrease the latency present in the processing of information from public sources, as well as decrease the load on MITRE itself, and provide the possibility of preventing damaging, uncontrolled bursts of vulnerability disclosures. The delay between MITRE and ICAT can be significantly reduced by the CVE Change Log mechanism of Cassandra, and NIST is planning to use it starting March 2002. We conclude that the usefulness of Cassandra will depend on having more people reserving candidates before publishing advisories, on the adoption of the CVE Change Logs mechanism by NIST, on the continued governmental support of the ICAT project, and on the excellent work done by the MITRE and ICAT teams.

5. References

[Arb00] Arbaugh, William A., Fithen, William L. and McHugh, John (2000) Windows of Vulnerability: A Case Study Analysis, IEEE Computer, vol. 33, no. 12, pages 52 - 59, December 2000.

[Car98] Carter, Gerald (1998) Patch32 : A System for Automated Client OS Updates. In: Proceedings of the Large Installation System Administration of

[8] Even a generic Poisson process will do this, given a sufficiently high rate of event arrival.

Windows NT Conference, 1998, Seattle, Washington

[CER01] CERT/CC (2001) Trends in Denial of Service Attack Technology: October 19, 2001.

[Liu00] Liu, C.; Richardson, D.J., (2000) Automated security checking and patching using TestTalk. In: The Fifteenth IEEE International Conference on Automated Software Engineering. Page(s): 261 -264

[Man99] Mann, D. E. and Christey, S. M. (1999) Towards a Common Enumeration of Vulnerabilities. In: Proceedings of the 2nd Workshop on Research with Security Vulnerability Databases.

[Mar02] Martin, R.; Christey, S. and Baker D. (2002) A Progress Report on the CVE Initiative. In: Proceedings of the 14th Annual Computer Security Incident Handling Conference (FIRST).
.

6. Acknowledgments

Thanks to Peter Mell of NIST for making ICAT available to us, and to Steve Christey and the CVE content team for generating the CVE candidates, without which none of this would be possible.

We gratefully acknowledge the support of the various sponsors of CERIAS for underwriting this work.