Case Study: "The Worm – Episode 1"

The Beginning

A warm Tuesday Afternoon, 2:00PM; the team had just finished lunch, and settled back to work. A phone rang, and was promptly answered. It was a senior official in Charge of IT at a Government Department, which delivers a very important service to the citizens of Sri Lanka. The Department had an Application Server running an extremely Critical Application, which processed hundreds of requests a day. We were informed that the Critical Application had become non-operational a few days earlier, and that a large number of service recipients had begun to queue outside the Office as a result. This event had begun to affect the productivity of the Department due to non-availability of a critical service; an indication of the possibility that this might be an *incident*, but we needed further information to verify that this could not be just an *event* such as a software bug, or memory problem.

The IT team of the Department had identified the situation several days earlier and had attempted to resolve the issue using internal resources. As part of their response, they had used a virus removal tool on the Application Server to remove files which the Anti Virus software showed as being infected by a Virus, but the problem persisted. The situation had continued to deteriorate, and the senior management had decided to ask for external assistance. In any case, we were asked to investigate the situation, which had now become desperate, and provide a solution for it.

Preparing for action

Preparation is everything. It was very important that we remained calm, and planned our course of action for this specific case based on standard procedures.

First, we needed to verify that this was indeed an incident. We initially collected some basic information over the phone and via E-mail from the official who contacted us to establish a background for the affected Department, so as to establish contacts and escalation and reporting points within the Department.

Only we had identified an incident and the cause of that incident, would we plan a specific response.

Yet again, we had four critical questions to answer: Is this really an incident? ***What*** caused the incident? ***Why*** was it so persistent that simple removal would not eradicate it? ***How*** did it enter the system in the first place?

The first question was immediately answered when the presence of the CME-24 was detected on the Application Server, by the Department's IT team, under the alias *W32.Blackmal.E*.

We also needed to gather our own information from the victim site. It was now time for the SLCERT team to visit the incident site.

The Team goes in

Upon entering the site, our team gathered more information to help with the investigation. The Application Server, which was running on Windows NT4.0 Operating System, had several shared directories, which enabled access to the Critical Application from Host machines in the network. Most of the client machines had detected a virus infection in the Shared folder of the Application Server.

Analysis of the gathered information began to give us a clear picture of the Department network setup, and the potential weaknesses that may have led to the incident.

Search & Research

In parallel to the on-site investigation, our team also conducted research on the *W32.Blackmal.E* worm, which is formally and uniquely known as CME-24. Like all other viruses, it was given different aliases by different Anti Virus Vendors. The site http://cme.mitre.org/data/list.html#24 provides details of all the aliases given to CME-24. Technical analysis had revealed that on the third day of every month, the virus overwrites files with extensions *.doc,*. xls, *.mdb, *.mde, *.ppt, *.pps, *.zip, *.rar,*.pdf, *.psd and *.dmp. This was initially identified on 17th January 2006 and all the antivirus vendors had released signatures for this worm. We now had an in-depth understanding of how the CME-24 worked; its strengths and its weaknesses.

It seemed clear that the *cause* had entered and modified entries in the System Registry, and so was able to regenerate itself on system restart; this is exactly what CME-24 does. That answered the question as to why the virus persisted despite its continued removal by the Department's IT team. By comparing symptoms such as this in the victim site with the Signature features of the CME-24, we were able to establish it as the cause of the incident, answering our second question.

For your reference, more information on the worm, with comprehensive technical analysis is available at:
http://www.symantec.com/security_response/writeup.jsp?docid=2006-011712-2537-99&tabid=2 .

Cracks appear

Several problems became apparent as the on-site investigation continued. Our team discovered that:

- ✓ Unfortunately, the Department's network was not protected with up-to-date Antivirus software.
- ✓ There was no documented Security Policy in place governing such security related procedures and actions.
- ✓ Users were accessing the Application Server system using a common password.

These were a set of long term Planning & Preparation issues that the Department needed to address.

Applying a Bandage

As an immediate remedy, we recommended the following immediate steps be taken to contain and recover from the situation.

1. Remove all the machines from Department network, including the Application Server.
2. Install and run up-to-date Antivirus software on the Application Server first, followed by selected critical Client computers, which accessed the Application Server.
3. Connect those clients back to the network and start operations as soon as possible.
4. Connect other client computers one by one after running Anti Virus applications on them individually.
5. *CME-24 also has built-in backdoors and other components that would not be detected and removed by Antivirus applications. Therefore, we recommended a*

total system re-installation of the Application Server. Leaving backdoors in a system allows an attacker to return and possibly re-infect or damage the system.

Since the site was already non-operational at that time these tasks were performed, the Department's supporting vendors and IT team were able to disconnect all the machines from the network, install Antivirus software, cleanse the machines and add them back to the Department network. The Antivirus vendor cooperated fully with the Department's IT team and our team to complete the removal and recovery jobs. This goes to show that working together maximizes the effectiveness of a response to an incident.

Lessons learned

This Government Department has been very fortunate to survive a major catastrophe. But some damage did occur in the form of lost service to its customers, which could have been minimized, had we been alerted at the very start of the signs of trouble. However, credit must be given to senior management for having the sense and the humility to obtain assistance from outside experts before the problem persisted any longer.

Generally, this incident could have been avoided if the Department had a ***Security Policy*** in place that was properly enforced, which for example, ensured that Anti Virus & Operating System software were regularly updated, Access to critical systems was restricted using a strong password policy and clear audit trails were maintained using system access logs. That's what makes Long-term Security Planning so important!

We submitted a report with this other recommendations to the Department.

The fourth question, How did the worm enter the system, is not completely answered here. Reconstructing the sequence of events which led to an incident is an involved process which is not done for incidents unless specifically requested by the victim of an incident. This area is called ***Digital Forensics***. In this case, it was not requested.

Case closed.