Global Information Assurance Certification Paper

Emanuel worm incident handling report
Vivian S. Burns

Executive Summary

This report describes an incident involving malicious code, specifically, the Emanuel variation of the Navidad worm. In the incident described here, the worm infected two desktop computers, rendering them unusable. The incident took place in the legal department of a large organization. The legal department has its own information systems personnel, who are responsible for the security of several departmental servers and the desktop computers.

This report describes our handling of this incident in terms of the six phases used in SANS incident handling training. I begin by explaining the phases and how they relate to the incident I am describing. Then I report in detail how we assessed and contained the worm, how we backed up the system, and how we preserved evidence.

Previous to the incident described in this report, a significant amount of incident handling preparation had taken place. We had many of the necessary policies and procedures, a designated incident handling team, and some of the necessary software and hardware. However, education on incident identification needs to be improved. The incident under review here was not identified soon enough. The result was that several personnel handled the system without consideration for preservation of evidence. Also, the antivirus software we were using did not recognize the worm.

The Emanuel worm was first discovered when a member of the office found her computer unusable one morning. None of her programs would run. After several attempts to resolve the problem herself, she contacted the help desk. They attempted to troubleshoot the problem, and rebooted the computer several times before contacting the incident handler.

The incident handler pulled the computer off the network and backed it up using DriveImage. She obtained information about the worm from the Internet. She contacted the person who sent the worm. She sent a warning about the worm to the legal office personnel. Help desk personnel went to all computers that had received a copy of the infected email to remove the executable. In the meantime, another user reported an infected computer, and we pulled it off the network.

Since the virus software that we use could not remove the Emanuel worm, we removed it by hand. We verified that each computer was working before replacing it on the desktop. Since then we have been monitoring for a recurrence, which has not happened.

As a result of this experience, we will be focusing on improving our incident identification procedures. A key element will be to educate users and help desk personnel as to what to

look for. We will also be switching to an antivirus vendor that publishes updates more frequently.

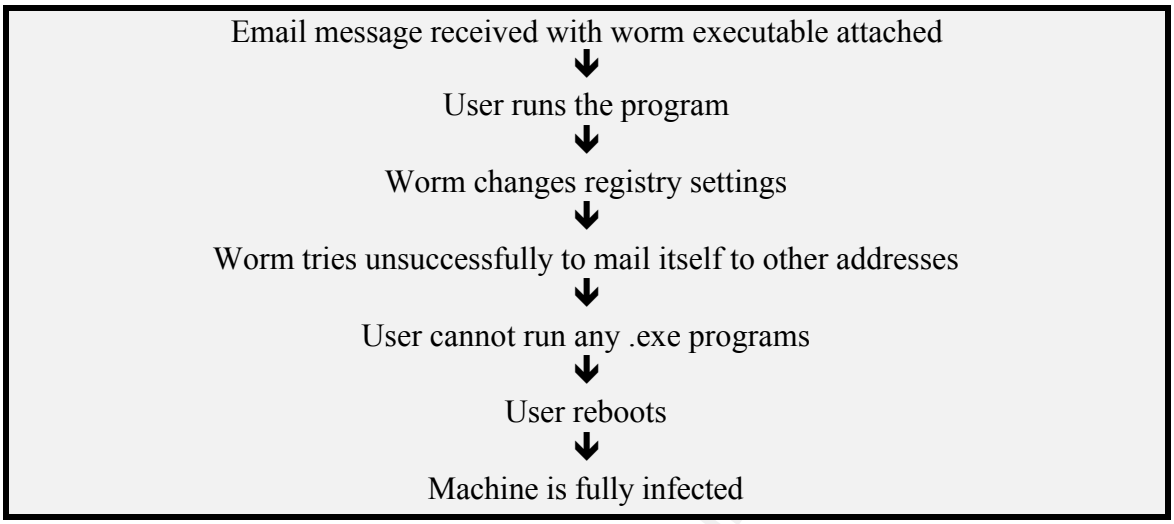


Figure 1. Sequence of events beginning with worm arrival

The six stages of incident handling

Phase 1: Preparation

The preparation phase can be seen as the most important stage of incident handling. It is distinct from regular security precautions in its purpose: to make incident handling more effective. Incident handling preparation supplements normal security precautions, which intend to reduce the rate of incident occurrence. The thoroughness of incident handling preparation will have a direct effect on the organization's capability to perform the important tasks of the other stages. In our case, the legal department had been making these preparations for some time before the incident occurred, but still had a lot left to do. The following paragraphs describe preparations needed and what we did.

Policy:

The policy aspect of preparation for incident handling requires the following:

1) That servers be configured with warning banners to the effect that the organization requires authorization for the use of the systems. The banners establish a legal ground for proving that an intruder was aware s/he was engaged in unauthorized activity. We had already configured such banners prior to the incident.
2) Policy on privacy. Privacy policy should clarify what information is off limits when using the system. In our case, the larger organization has formidably lengthy policies on information security and privacy. We wrote and disseminated a short version of the policies for the department's use, containing relevant points.
3) Policy on the notification of law enforcement during an incident. It is also helpful to find out ahead of time which cases interest law enforcement.
4) Policy on "contain and clear" or "watch and learn." The question is, does the

organization prefer to find out exactly what an intruder is doing, or does it prefer to get the intruder out of the system as quickly as possible? Our organization has not stated a policy on this point. As a department, we responded to the lack of policy in our Emergency Action Plan packet, described below. The packet states that departmental priority in an incident is to "contain and clear." We want our people back to work as soon as possible.

5) Policy on additional authority for incident handlers during an incident. The department has not finished its considerations on authority. In cases that we are aware of, the incident handler already has authority in her job description to make decisions regarding the system.
6) Policy on whom to notify about incident occurrences. We included these phone numbers in our Emergency Action Plan packet.
7) Policy for handling incidents on remote computers. For us, remote computers are not a big issue because we provide neither remote access nor laptops for our people.

People:

The people aspect of preparation for incident handling requires the following:

1) That appropriate personnel be designated, organized and trained as incident handlers. Since the larger organization has no incident response handling team, the legal department has designated the full-time members of our technical staff as our incident response team. We selected these team members based on their technical expertise and the fact that they are the only members of the department that work full time in technical positions. The help desk staff do not handle incidents. The senior technical person, who attended SANS incident handling training, is the incident team lead.
2) That personnel be sufficiently educated concerning security. There must be

- An understanding of the objectives of incident handling. Since the person reporting a problem is usually not a trained incident handler, it's a good idea to publish a list of signs that an incident may be taking place.
- Financial support for necessary resources, such as those mentioned below.
- Cooperation among relevant personnel, such as help desk, system administrators and CIRTs. Incident practice drills can be useful.

Our efforts in this area are comparatively new to this office, so we do not yet have strong support. We are working on improving it.

Data:

For a team to be effective during an incident, their preparation should include gathering information needed during an incident, such as phone numbers, passwords, and encryption keys. We used the SANS Emergency Action Plan as a basis for a Computer Emergency Action Plan that we revised to suit this office. We assembled relevant phone numbers for emergency contact personnel. Then we assembled the EAP, the phone numbers and copies of SANS incident handling forms into a set and made several copies. We located sets of forms where members of the incident handling team can get them easily.

Software and Hardware

Incident handling teams need software and hardware tools to investigate an incident. Some of the tools that can be useful are:

1. Regular backups previous to the incident. These will enable handlers to restore corrupted data when necessary. Accordingly, we run a *verified* full backup to tape on all servers every night. We take tapes off site weekly. We run restores fairly frequently to be sure the backup and restore process works. For desktop computers, we have an automatic backup to server that takes place daily for certain specified directories. We keep most documents in a repository on a server.
2. Backup tools. The best backup tools for use during incident handling will do a bit-by-bit backup. A bit-by-bit backup enables copying "empty" spaces on a hard drive that may contain erased data. We do not yet have such a tool, so we have requested one. In the meantime, we use a hard drive that plugs into the parallel port of any computer. We boot from a floppy to use this hard disk, which loads a disk imaging tool, our interim backup tool for incident handling. We have all of the following:
3. A tape recorder used for recording actions taken during an incident.
4. A small hub used for maintaining a network connection when a handler removes a computer from the main network.
5. A dual OS laptop for running forensic software, and forensic software. The laptop attaches to the hub in order to run diagnoses on the afflicted system.
6. Cell phone for each team member.

Communication, Transportation and other aspects of infrastructure

Infrastructure support during incidents is critical. An organization needs to plan how communication lines work during an incident, how incident handlers will get to the scene, where the command center will be, how power will be supplied, etc.

For team communications during an incident, we have a simple plan. Help desk personnel must notify the senior technician / lead incident handler of any suspected incident. If the incident will affect more than 10% of the users, she notifies the director. Team members carry cell phones, programmed with the relevant numbers. For reporting of incidents, users may email or phone the help desk.

Other infrastructure issues we leave to the larger organization.

Phase 2: Identification

During the identification phase, the following take place:

1. Someone reports a suspected incident, preferably according to established procedure. Encourage personnel to err on the side of reporting early, rather than waiting for certainty. A false alarm is usually less costly to an organization than an unreported incident or one that is caught late.
2. Next, the designated incident handling authority assigns a primary incident handler to determine whether the event is an incident, or just a bug, mistake, or some other type

of problem. An incident implies *harm or the intent to harm.*

Phase 3: Containment

During the containment phase, the team takes steps to keep the problem from getting worse. They will need to:

1. Stay calm. In a high-stress situation such as an incident, using the mind well is especially important, because there is plenty of potential for making mistakes that may make the situation worse.
2. Take notes. Best practice is to have at least one additional team member accompany the primary handler. Having an additional person on hand enables the team to maintain proper communications and records.
3. Secure the area. Keeping the system in pristine condition is important for preserving evidence. Therefore, to the extent possible, the team should keep the system in the condition they found it in. Handlers should prevent work on the system until they have resolved the incident. They should also secure the system physically. Handlers should record details of the affected system. They should make a complete record of events reported to have happened before their arrival
4. To keep the problem from spreading, consider pulling the system off the network. The handler can use a small hub to maintain the network connection, thereby minimizing changes to the system.
5. Back up the system. Best practice is to run a bit-by-bit backup from a separate boot disk or from a laptop attached to the portable hub. As previously mentioned, bit-by-bit backups enable a handler to capture deleted information that may still be present on the disk. The best procedure is that the handler should make two backups. The team can keep one backup untouched to maintain the chain of evidence, and use the other for forensics. The most preferable, if possible, is to store the original hard drive itself as evidence. Whether your evidence backup is the hard drive or a copied backup, label it and store it in a place where no one will disturb it. Load the forensics backup onto a clean system for analysis.
6. Decide whether to keep the system running or take it down (if you haven't already pulled it off the network). The handler needs to consider what other systems have trust relationships with the affected system, as well as the criticality of the system to the organization.
7. Change passwords on the affected system and any systems that trust it. This will prevent spread of the damage by disabling an intruder from using passwords they may have discovered.

I describe how we assessed and contained the Emanuel worm incident in the third section of this report. The fourth section describes the backup procedure we used.

Phase 4: Eradication

The team repairs the system during the eradication phase of incident handling. It includes:

1. Determining the cause of the incident. This may be the most difficult part of the whole process. Because the number of vulnerabilities and exploits is huge and growing, anyone working in computer security must constantly improve their awareness of

these exploits and of how to detect and diagnose them. Examples of some clues a handler can look for are: suspicious network traffic, suspicious logons, unusual log entries, and unexpected files. Good notes about actions taken and results observed are often useful during diagnosis.

2. Once the team determines the cause of the problem, they must improve the defenses that may have allowed the incident to happen. For example, if an intruder has compromised a firewall using a known vulnerability on a server, the team should see to it that someone patches or reconfigures that server.
3. The team should perform a vulnerability analysis on the affected system and possibly nearby systems. Incorrect security configuration may have caused the compromise. In that case, additional vulnerabilities may be present. Other systems may have been misconfigured as well.
4. Then the handler removes the cause of the incident. A handler may remove a problem before improving the defenses or performing a vulnerability analysis, but doing so will leave the system open to the same exploit.
5. Then the team needs to locate a system backup in preparation for the recovery phase.

Phase 5: Recovery

During the recovery phase, the incident handling team brings the system back into normal operation. This phase includes:

1. Restoring the system and / or data from backup if necessary. When an incident takes place, the system or data may be corrupted or lost. Good preparation includes regular backup so that when the cause of a problem has been cleaned up, the data and programs on a system may be restored to their previous state. In cases where the extent of corruption or infection is unclear, the handler should consider wiping the disk and doing a fresh install.
2. Before putting the system back into service, the handler should validate that the system is clean. Double-checking that the problem has been completely removed prevents a scenario where the team comes back again and again to work on the same problem.
3. Next, the team restores operations to normal
4. And last, they will continue to monitor for re-infection. An intruder sometimes gains information that makes a second intrusion easier. In our case, we have been monitoring for recurrence of this worm for several weeks since and have seen none.

Phase 6: Follow-up / Lessons learned

The follow-up phase of incident handling includes:

1. Reporting. Taking complete notes during the entire process makes this step much easier. Best practice, as mentioned previously, is that there should always be at least two team members on hand so that one can be responsible for good recordkeeping and communications. The report should present information about the affected system(s), observed symptoms, actions taken, and decisions made.
2. Conducting a Lessons Learned meeting. In this meeting, all team members have the opportunity to comment on the report. The team reviews the entire incident and how they handled it. They look for ways that they could have performed better, and ways

that the organization may prevent similar incidents. They make proposals for how they will perform better the next time.
3. Writing an executive summary of their proposals and sending it to management.
4. Implementing any improvements that management approves.

Assessing and containing the Emanuel worm

The incident handlers first became aware that a problem existed when a help desk technician contacted them to report it. The handlers went to the location to look at the problem, and asked for a description of events to that point. This is what had happened:

The incident began with problems observed at a user desktop. The user had booted up her computer as usual, checked her email, and then tried to run Microsoft Access. Instead of the expected program, she observed a message box. The "Error" message box contained the characters ";)" When she clicked the "OK" button, she received another error message concerning an illegal operations error by "Wintask.exe." She decided to reboot in the hope that the problem would disappear on its own. When she did, she expected to see several programs starting automatically, as they do every day. Instead, she saw the same smiley face messages, followed by illegal operations messages. She then attempted to start her email program manually, using a shortcut on the desktop. The same message box appeared again. She next right-clicked on the shortcut and chose an anti-virus scan. The anti-virus software started up and scanned the program. The scan reported no viruses. She then selected the "send" menu option in the antivirus software, which uses mapi to email virus reports. This action successfully opened her email program.

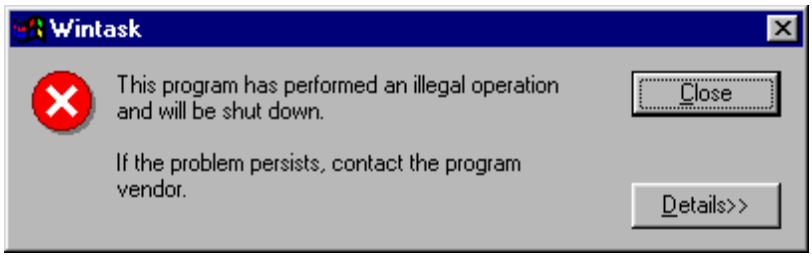
The user reported her difficulty to the help desk. A member of the help desk team went to the location and attempted to run the antivirus software from a command line. The antivirus software, however, would not start. Instead the same message box appeared.

Then the help desk technician asked another technician for help. The second technician right-clicked on the hard drive icon to select a virus scan. The virus scanning software successfully started and scanned the entire hard drive without finding any viruses.

The first technician tried to run scandisk from the menu. He received the same error message. He rebooted. All of the startup applications again failed, and again the message boxes appeared. He rebooted the computer again. At this point the technician escalated the problem to the incident handling team.

When the team arrived, the user demonstrated the problem to the handler by again trying to start the email program from a desktop icon, resulting in the same error messages:

and



When the handler saw the symptoms that the user was demonstrating, she surmised that the computer may have been experiencing a virus infection. She pulled the network cable from the back of the machine and began recording information.

The computer was running Windows 95, like most of the other desktops in the legal department. It was a four-year-old Compaq that logged in to several different server operating systems every day. Since the system was a desktop system, we could provide the user with a clean replacement system while we determined the cause of the problem. We told her she would have to make use of data from the previous day's backup until we could secure this system.

As we were about to shut down the computer, the user mentioned that she had received an unusual email the previous day. It contained an attachment, which she had opened. She said it didn't appear to do anything and she didn't think anything of it at the time. When she showed us the message in question, we saw that the attachment had been named "Emanuel74.exe".

The attachment itself was no longer present. The network login script deletes all .exe files in the email attachments directory every time a user logs on. Unfortunately, a user can still run an executable when it first arrives if she is so inclined.

We shut down the system and moved it from the user's work area to a locked room where we locate our forensic tools. Our tools include the following:

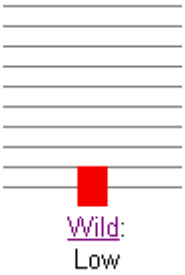
- Dictation machines
- An external BackPack hard disk that can attach to a parallel port
- Bootable floppy disks for use with the external hard disk
- The SANS disk from incident handling training
- A small hub
- A laptop
- And copies of the Computer Emergency Action Plan for the department.

After we moved the system, we backed it up using DriveImage software running from an external Backpack hard drive attached to the parallel port, as explained in the next section. The DriveImage software created an image that we copied to a clean system to investigate as needed. Whenever necessary, we could set the system back to the state in which we

received it.

The next step in assessment was to look for information about "Emanuel74.exe" on the Internet. The handler went first to the antivirus software vendor that we have been using. At the time, they had no useful information on their web site. However, upon expanding her search, she found the description of the Emanuel worm at web sites by Symantec and others. Symantec refers to the Emanuel worm as "W32.Navidad.16896." Here are some of the details that we learned from Symantec:

- The Emanuel variation first became known in November of 2000.
- At the time of our incident, its frequency in the wild was low.



- It does not cause extensive damage to the affected system. It alters a few registry keys and installs one file.
- The worm spreads via Microsoft Outlook

Here is how the worm works:

- The worm arrives as Emanuel.exe
- When executed,
    - It displays an "Error" dialog box with a smiley face.
    - It copies itself to c:\windows\system\wintask.exe.
    - It adds registry settings that take effect the next time the computer boots. The registry settings are detailed below.
    - It uses mapi to send replies to each message in the inbox that has one attachment. It attaches itself to each reply.
    - It installs a flower icon in the system tray.  Further behaviors are expected upon interaction with the icon.
- Upon reboot, one of the settings causes the Wintask program to run automatically.
- Another setting makes the Wintask executable the shell command for the ".exe" type. This means that whenever a program file with the "exe" extension tried to run, the system would instead run the Wintask program. Since the setting only applies to one type of extension, "com" or "bat" files could still run.

Four aspects of our infected system differed from the description:

- The executable file had been named slightly differently than expected.
- According to the user, there were no observable symptoms upon first execution, not even the smiley.
- The system was consistently producing illegal operations errors following the smiley

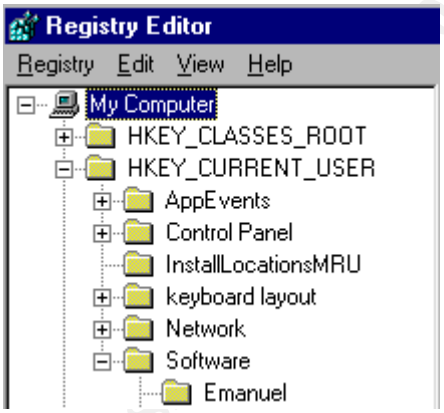
dialog, something that the worm description did not mention.
- At no point did we observe a flower icon in the system tray.

The email header of the infected message showed that copies had been mailed to several other people in the office. We decided to check the computers of all the users listed in the header. We enlisted the help desk technicians for this check, explaining that if they found the executable they were to delete it and report back. Within a few minutes, they had completed the check. They found no copies of the executable. We hoped that this was the only infected computer.

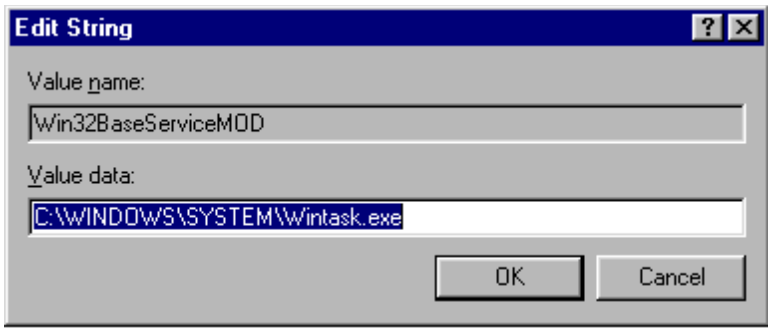
Meanwhile, the lead incident handler called the person who had sent the infected message to notify her that we had received a virus from her. The handler also sent an email to the entire department, warning that an infection had taken place, and reminding them of email handling precautions. The purpose of this email was mainly educational. She had previously warned the computer users in the office that running unknown programs received via email can be dangerous. Some computer users ignored the warnings -- our infected system was proof of that.

Continuing our examination of the affected system, the handler checked the email outbox for infected messages. She found none, another difference from the Symantec description.

An inspection of the registry yielded these results:
- The worm had added an "Emanuel" key to *HKCU\Software*:



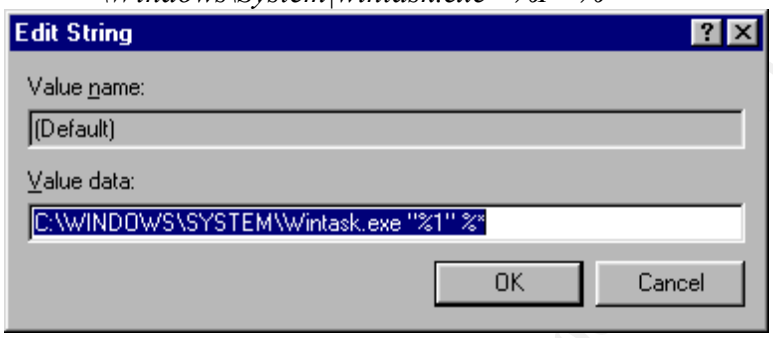
- The worm had added Wintask to the key *HKLM\Software\Microsoft\Windows\Current\Version\Run*.



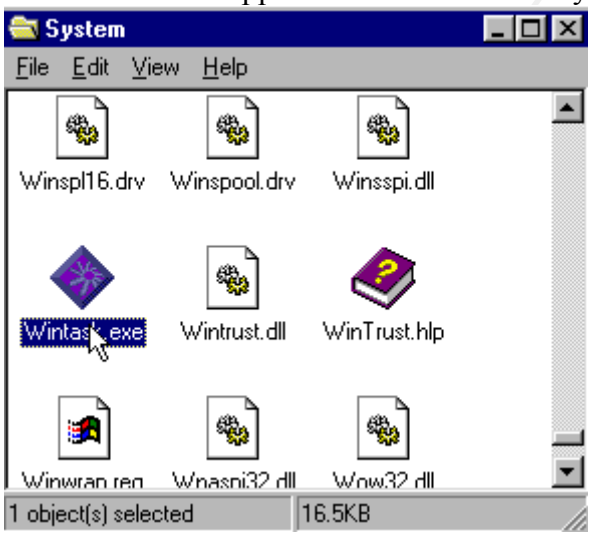
- The worm had modified the following key:
  HKLM\Software\Classes\exefile\shell\open\command

with the value:

*\Windows\System|wintask.exe "%1" %**"



The worm had dropped Wintask.exe in the system directory.



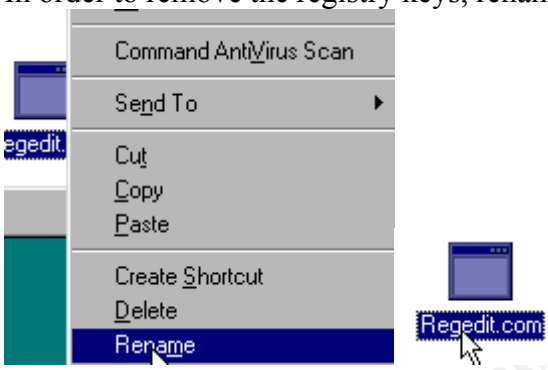
The handler's assessment was that either we were dealing with a close variant of the Emanuel version of the Navidad virus, or there were explanations for all of the observed differences in behavior:

- The difference in the executable name could have been caused by a mail program that adds digits to the end of files with duplicate names.
- The omission of the smiley face on first execution could have been an error of observation on the part of the user.

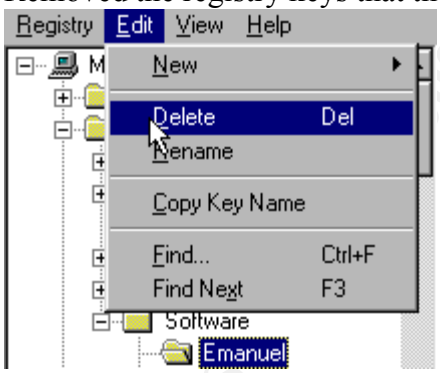
- The lack of outgoing messages may be accounted for by the absence of Microsoft Outlook on the system, since we use a different email program, which nonetheless does include mapi.
- The illegal operations error may have been caused by the absence of Microsoft Outlook on the infected system. Or, it may have been caused by the absence of certain other standard Windows files, which we had removed for security reasons.
- The absence of the flower icon may have been caused by the program crash prior to finishing its routine.

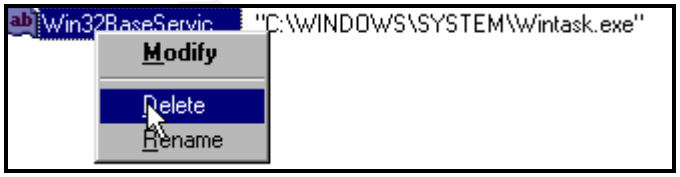
To find out which of the two alternatives was the truth, the handler carried out some experiments. First, she brought the system back to its pre-infected state:

- In order to remove the registry keys, renamed regedit.exe to regedit.com.

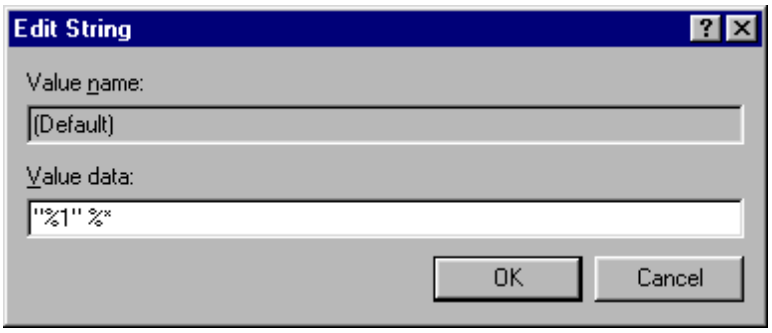


- Removed the registry keys that the worm had added.





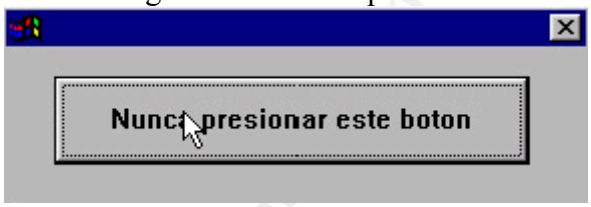
- Changed the modified key back to its default value: *"%1" %**"



- Copied Wintask back into the email directory under the name Emanuel74.exe.

Investigation proceeded as follows:

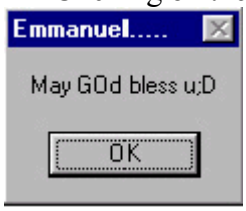
- She ran the worm executable from the email message and observed the same smiley face and illegal operations error as before. This result is different from what the user had observed.
- She set the registry keys back to normal again. Then she installed Outlook Express and ran the worm executable again. She observed the same result.
- She ran the worm on a system that had a normal installation of Windows. On this system, the executable behaved exactly as described by Symantec:
    - No illegal operations error appeared after the smiley.
    - A flower appeared in the system tray.
    - Placing a cursor over the flower resulted in the message:
      Come on lets party!!!
    - Clicking the flower icon produced the following:



(Never press this button)

    - Clicking on the button produced the following:

    - Clicking on the 'X' to close the "Nunca" window produces:



The investigator concluded that the worm we had was the same one described even though the executable name was slightly different.

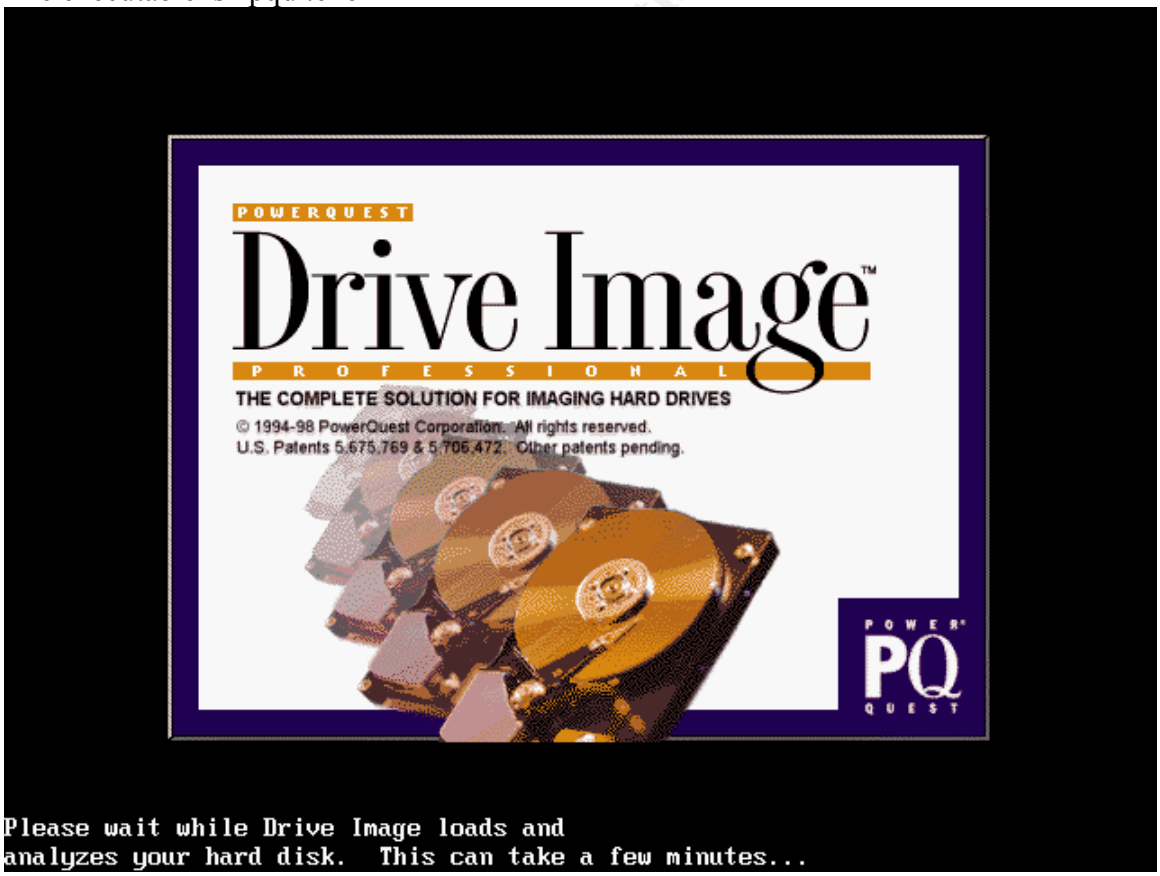
While we were investigating the first infected system, another user reported that her computer was not working. Upon arrival at the scene, the handler saw that the user was experiencing the same symptoms the team had seen previously. The handler told the user that the system had a virus that she needed to remove. The handlers disconnected the machine from the network, shut it down, and took it to the lab. She removed the virus using the steps outlined above, except she deleted Wintask. The machine appeared to run satisfactorily when these steps had been completed.

Later, further information about the worm appeared at the vendor web site and they included it in a definitions update. At this writing all of the major antivirus vendors had information about the worm on their sites. For web site addresses, see end notes.

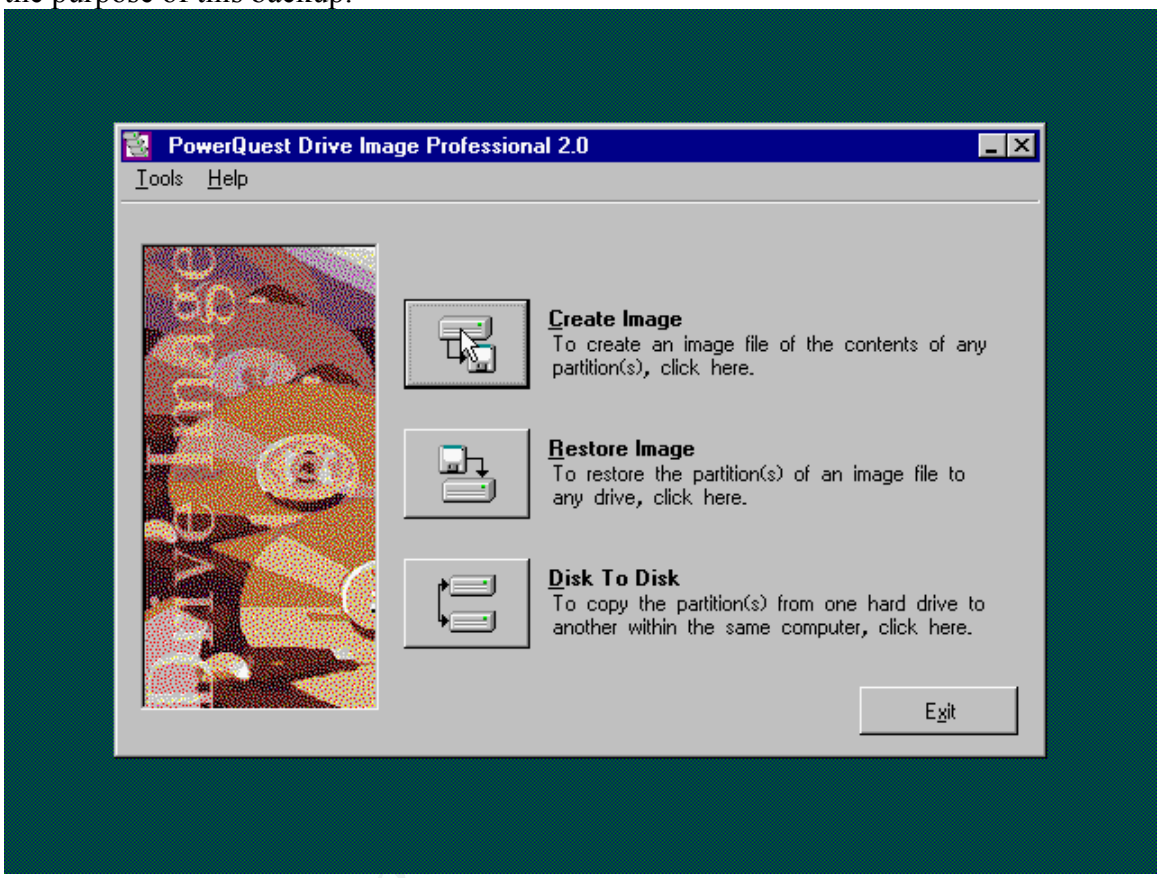
Process used to backup system

The incident handling team does not yet have a bit-by-bit backup tool. Therefore, we used the next best tool that we had. This is the process that we used:

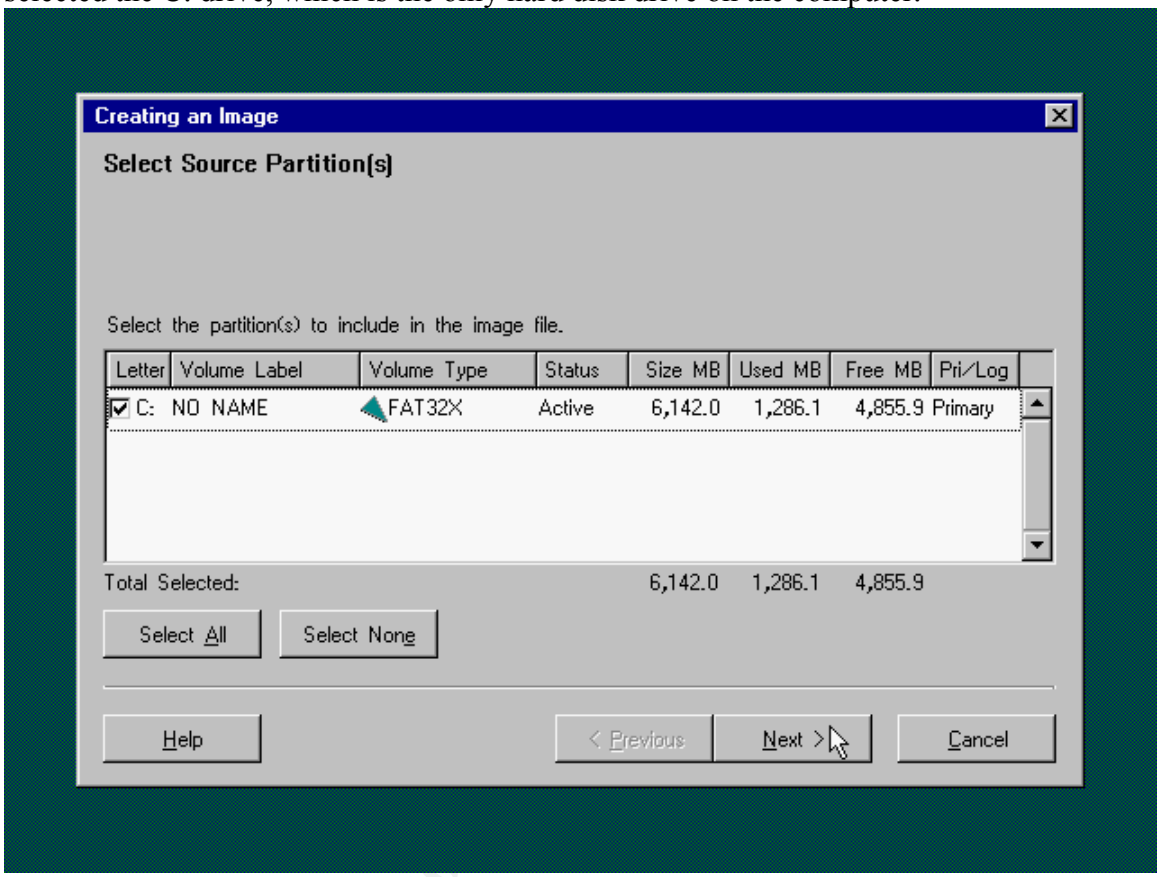
1) Connected an external Backpack drive to the parallel port of the infected system. We keep the drive in our lab at all times.
2) Inserted a write-protected bootable disk with the Backpack driver into the floppy drive. We prepared the disk ahead of time and keep it with the Backpack drive.
3) Booted the computer. The config.sys file loads the Backpack driver:
   `device=bphddrv.sys`
   which enables the system to recognize the four partitions present on the drive. The autoexec.bat file automatically runs the DriveImage program, which is present on the second partition of the drive:
   e:
   cd \drvimg
   pqdi
   The executable is "pqdi.exe"



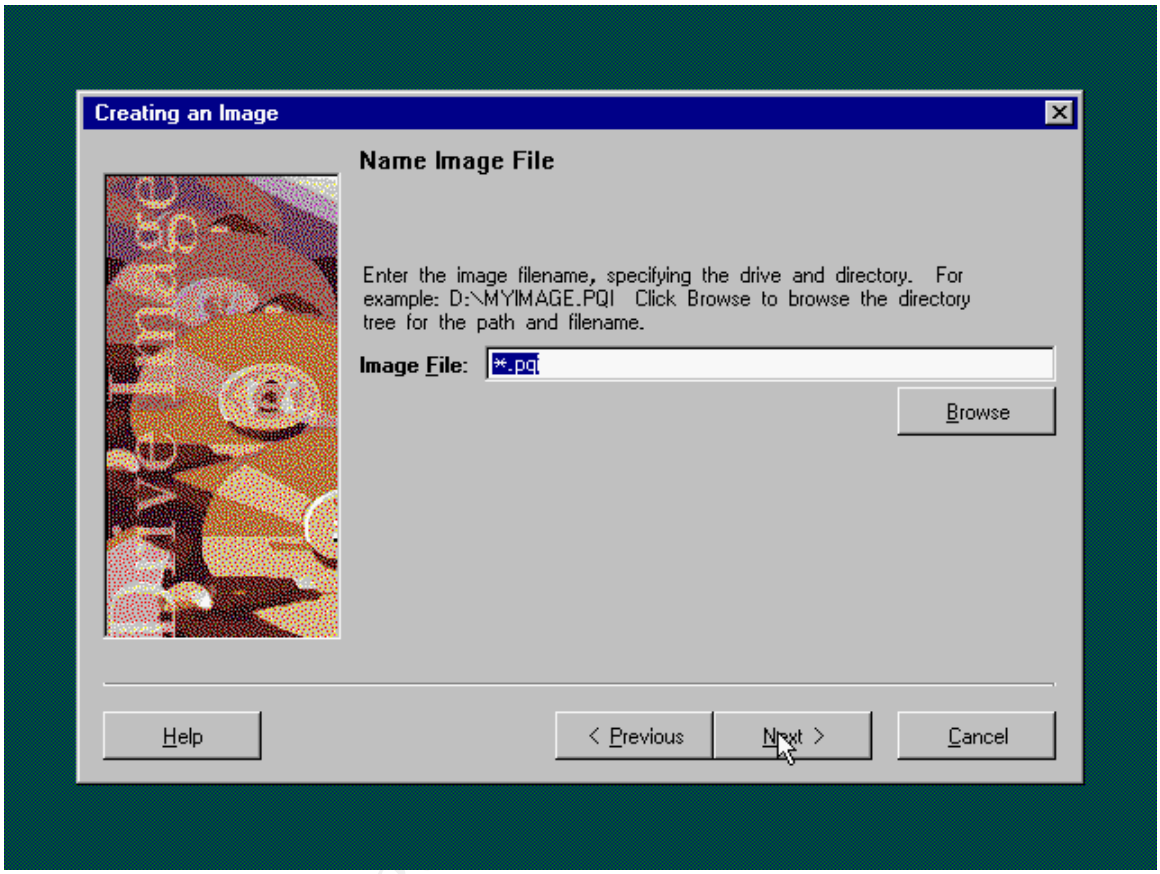
4) When the software finishes loading, it presents a selection screen. The choices: “Create Image”, “Restore Image”, or “Disk to Disk”. We chose “Create Image” for the purpose of this backup.



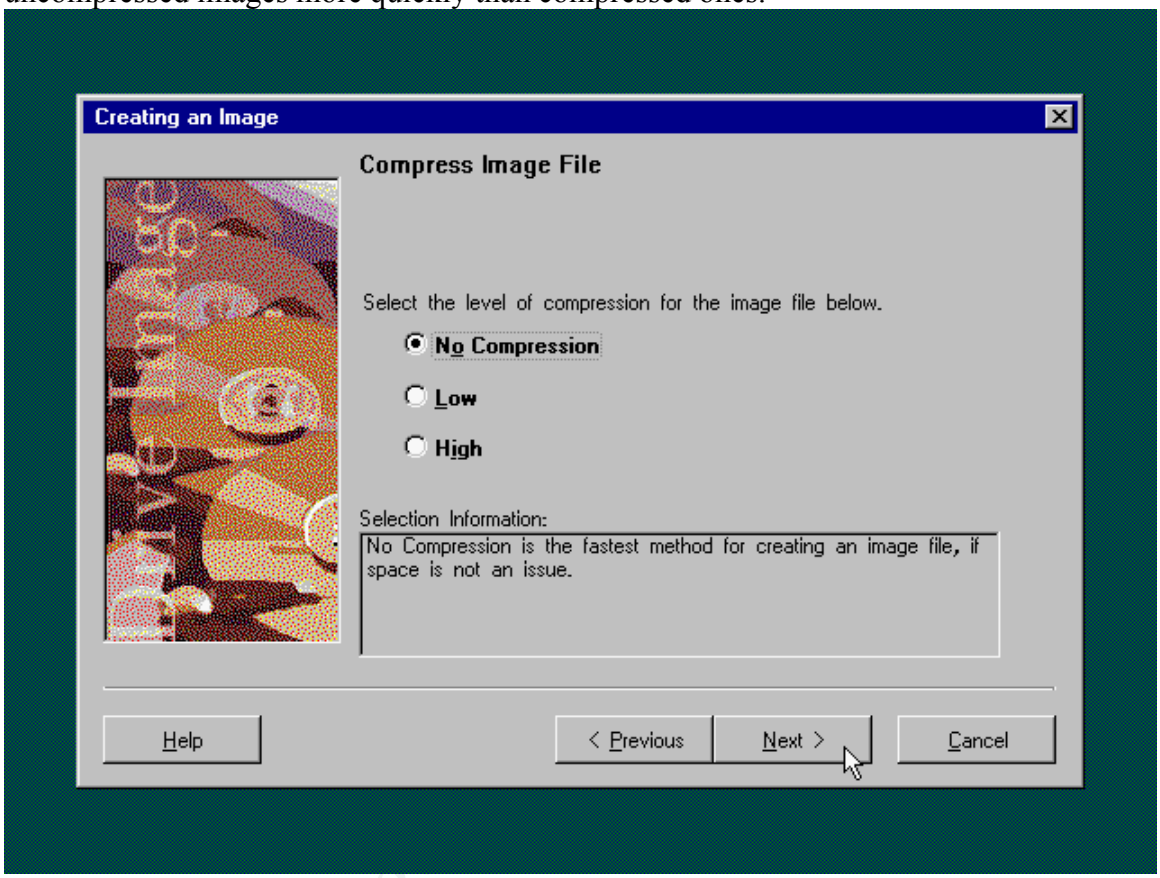
5) Next, the software needs to know what disk will be the source of the image. We selected the C: drive, which is the only hard disk drive on the computer.



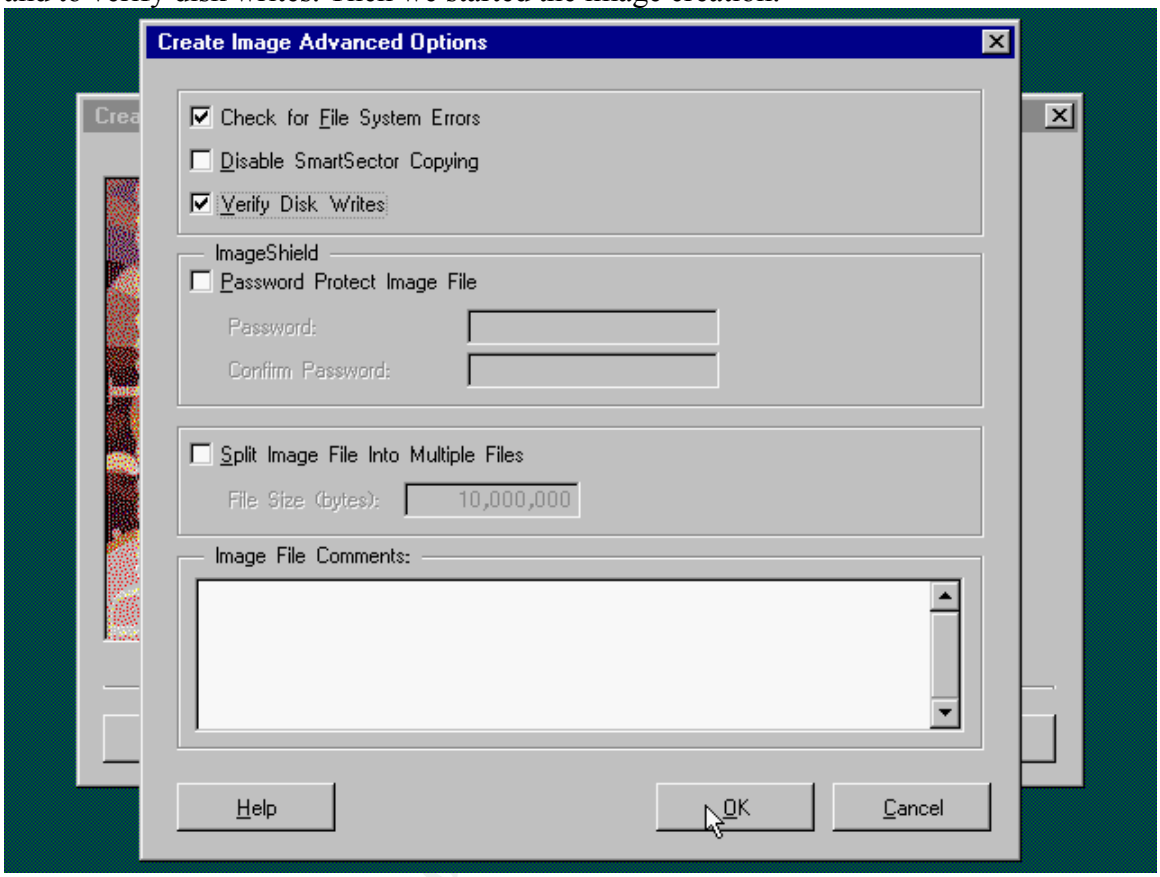
6) Then the software needs to know where to put the image it creates. We saved the image on the Backpack drive under the name “Navidad.pqi”.



7) The software allows a choice of compression. We did not compress our image at all, so as to minimize the number of problems that could occur. The software also makes uncompressed images more quickly than compressed ones.



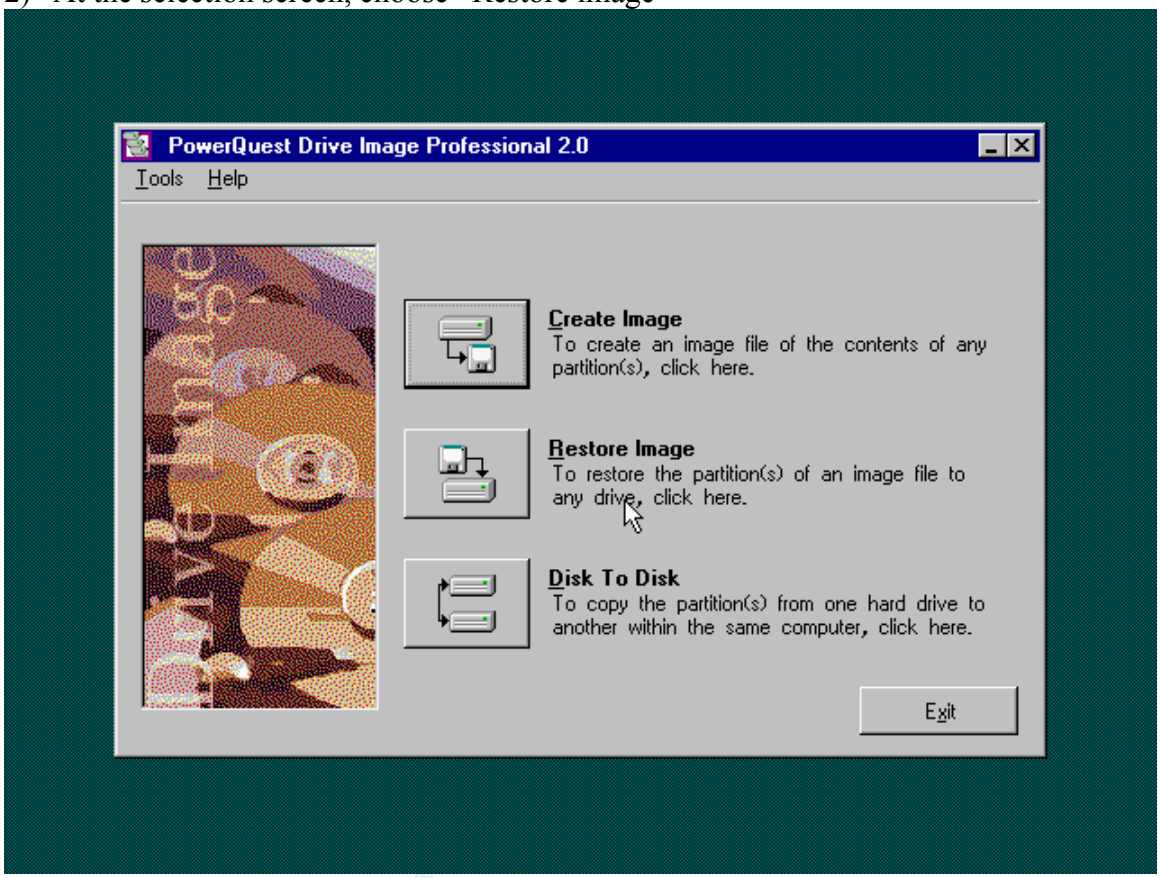
8) In the “advanced” options, we chose to have the software check for file system errors and to verify disk writes. Then we started the image creation.



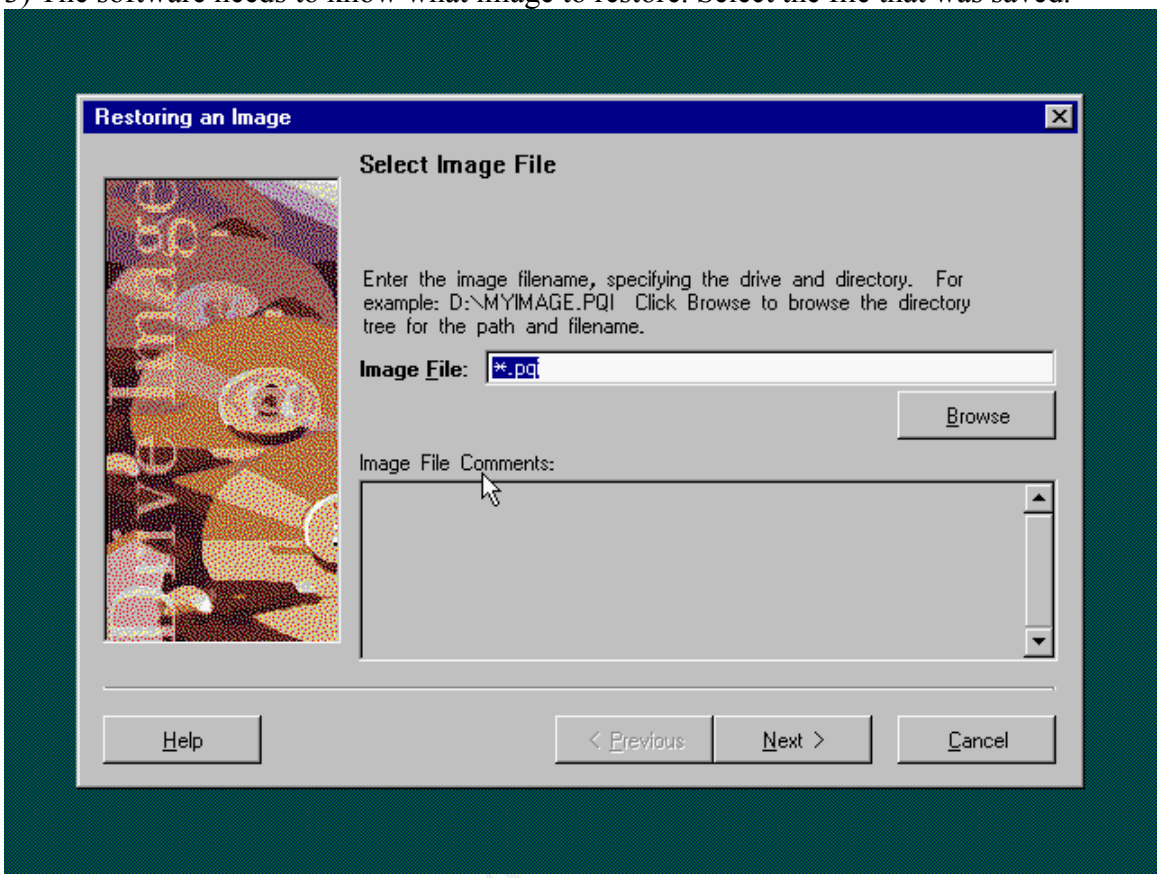
9) The software created the image. We could now restore it to any hard disk drive for use in our investigation.

To continue our investigation, we restored the saved image to a clean system. Here is the procedure we used:

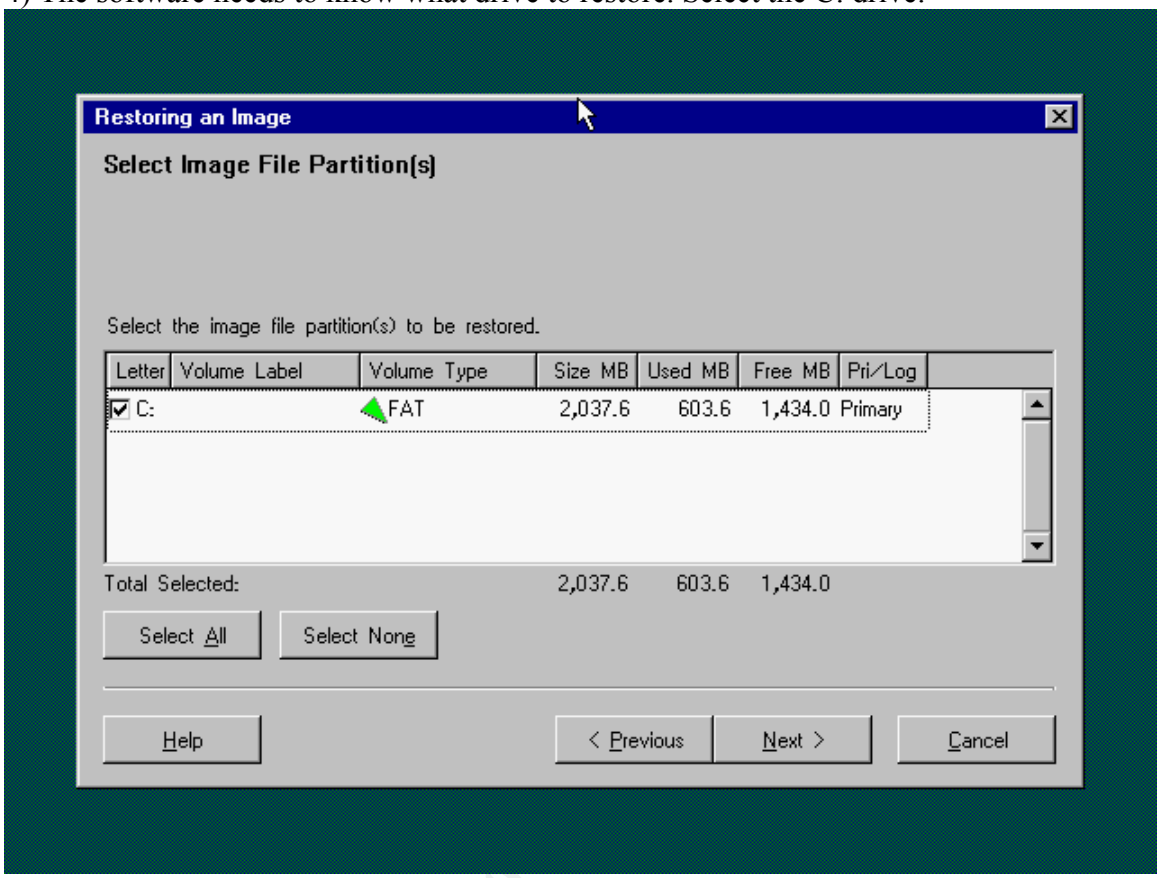
1) Connect the BackPack and boot from a floppy as before.
2) At the selection screen, choose “Restore image”



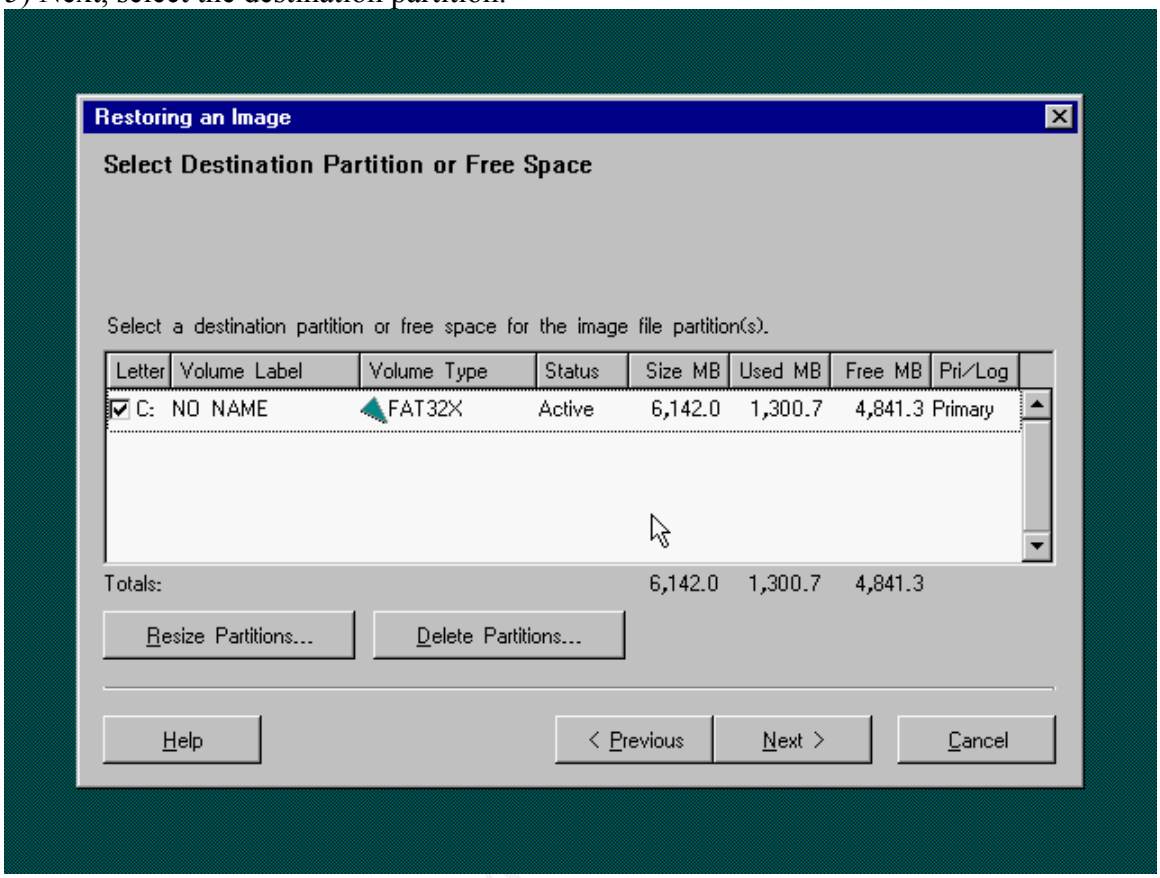
3) The software needs to know what image to restore. Select the file that was saved.



4) The software needs to know what drive to restore. Select the C: drive.



5) Next, select the destination partition.



6) Run the restore, reboot and you're ready to go.

Chain of custody procedures

In order to prevent corruption of the evidence as much as possible, we did the following:

- As soon as the incident handling team had been called to the site, we requested that the user stop using the computer.
- We removed the network cable from the computer. This prevents the computer from being modified over the network.
- We moved the computer to a locked room controlled by incident handling personnel.
- We made a backup of the hard drive as it existed at that moment.
- We made use of a separate system for our investigation of the worm.

Listing of evidence:

- The drive image of the system as it existed at the time we received it. Includes:
  - Copy of the worm executable
  - Copy of the email message that contained the executable
  - Registry settings altered by the worm
- Notes made on what procedures we followed, and when.

End notes

As of 1/24/01, sites with information about the Emanuel variation of the Navidad worm:
http://www.symantec.com/avcenter/venc/data/w32.navidad.16896.html
http://vil.mcafee.com/dispVirus.asp?virus_k=98881&
http://www.antivirus.com/vinfo/virusencyclo/default5.asp?VName=TROJ_NAVIDAD.E&VSect=T
http://www.europe.f-secure.com/v-descs/navidad.shtml
http://www.sophos.com/virusinfo/analyses/w32navidadb.html
http://www.norman.com/virus_info/w32_navidad_b.shtml

Many of the screen shots included in this report were not taken at the time of the incident, but are included for clarity only.