



WHITE PAPER
TRENDLABS RESEARCH

JUNE 2004

TREND MICRO, INC.
10101 N. DE ANZA BLVD.
CUPERTINO, CA 95014
T 80022.8.5651/408.257.1500
F 408.257.2003
WWW.TRENDMICRO.COM

The SASSER Event: History and Implications

TABLE OF CONTENTS

OVERVIEW 3

I. THE SASSER STORY 6

 The War of the Worms 6

 The Netsky-Sasser Connection 6

 Finding Sven 7

 Motives and Mind Games 8

 Friends and Foes 9

II. THE SASSER INFECTION ROUTINE 11

III. MSBLASTER and SASSER: Cut from the same cloth? 12

 Basic Exploit Similarities 13

 Infection Technique 14

 Symptoms 15

 SASSER: Fast, Not Sassy 17

IV. SASSER Variant Comparison 18

V. NETSKY and SASSER: Brothers in crime? 22

VI. The SASSER Bandwagon 24

VII. Implications of a SASSER 26

Appendix A: Sources 28

Appendix B: Detailed Timeline 29

June 2004
Trend Micro, Inc.

©2003 by Trend Micro Incorporated. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the prior written consent of Trend Micro Incorporated. Trend Micro, the t-ball logo, and TrendLabs, are trademarks or registered trademarks of Trend Micro Incorporated. All other company and/or product names may be trademarks or registered trademarks of their owners. Information contained in this document is subject to change without notice.

OVERVIEW

Analyzing malware should not be limited to outlining behavior and countermeasures. In an interconnected digital universe, every malware released into the wild brings about a myriad of possible repercussions. Thus, analysis becomes more complicated as virus researchers and security professionals worldwide find additional ways to look at each malware.

This paper is not an exhaustive technical guide on how SASSER operates and how to deal with it. Rather, it presents the said malware family as an event that has a unique context. Hence, this study is primarily concerned with SASSER's behavior in relation to other chronological events and other malware families.

The prolific SASSER family led TrendLabsSM into declaring a total of two virus alerts for the first quarter of 2004. WORM_SASSER.A, the original SASSER, was declared a medium risk virus on May 1, 2004. Hours after this declaration, a high-risk virus alert was issued for all existing SASSER variants.

As of June 10, 2004, this highly successful family of vulnerability-exploiting worms has spawned a total of five variants, the last of which was discovered on May 10, 2004. Section IV offers a more detailed discussion on the evolution of the SASSER worm.

WORM_SASSER.E, discovered on May 8, 2004, was considered as the most widespread variant of the family, with approximately 56,000 reported infections worldwide according to Trend MicroTM World Virus Tracking Center (WTC). This particular variant is currently ranked third in Trend Micro's Top 10 Virus Threats List.

One of TrendLabs' predictions for 2004 was that mass mailing and blended threats would continue to be the standard for hot malware. NETSKY is an example of a well-known blended threat. Besides propagating via email, it also propagates via shared networks. Some of its variants perform denial of service (DoS) attacks against a list of target Web sites. WORM_NETSKY.V even experiments with exploits. Details of this worm's evolution are covered in Section V.

Interestingly enough, SASSER totally eschews the virtue of social engineering to propagate. SASSER's strongest feature is that it exploits the WindowsTM LSASS vulnerability, a hole that allows remote code execution on an infected system. Read about SASSER's infection and propagation routine in Section II. What makes it even more interesting is the fact that when SASSER is juxtaposed with MSBLAST (commonly referred to as MSBLASTER, *the* Internet worm of August 2003), a similarity in behavior can readily be observed.

SASSER is undeniably the MSBLASTER event of 2004. Some key points of comparison between these malware families include the following:

1. attack via exploit (OS vulnerability)
2. employ propagation routines
3. attract media attention

Section III offers a more detailed discussion on the comparative analysis of SASSER and MSBLASTER.

Other malware authors hoping to extend the life and reach of their creations (while at the same time attempting to steal some of the spotlight from SASSER and its suspected author) have hatched worms that have routines directly connected to SASSER. These include CYCLE, DABBER and KORGO. Details on these malware can be found in Section VI.

SASSER does not have a malicious payload, meaning that neither destroys nor alters data on an infected system. However, its rapid propagation across unpatched machines can bring down networks and adversely affect business processes.

Cases of damage¹ included the following:

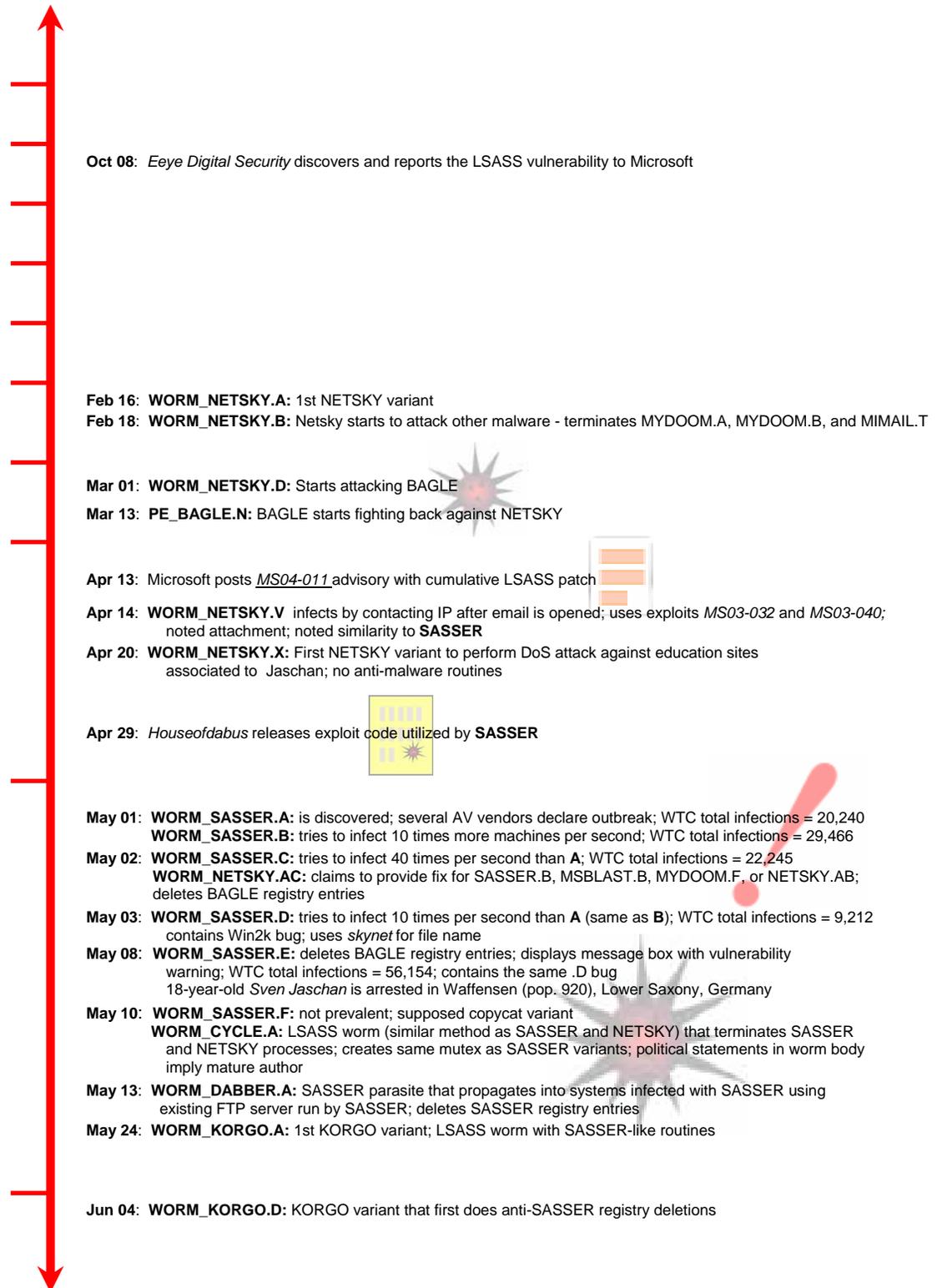
1. public hospitals in Hong Kong
2. one-third of Taiwan's post office branches
3. British Airways – 20 flights were delayed for 10 minutes
4. Sydney train system
5. Scandinavian banks
6. British Coast Guard – 19 control centers were forced to use traditional pen and paper for their charting routines.

These cases confirm that the effects of virus outbreaks are very real and that people from all walks of life suffer the damages. No matter what motive lies behind the creation of SASSER and no matter how people feel about its suspected author, the fact remains that considerable damage has been done to many. Section VII describes the lessons learned and conclusions drawn from the SASSER event.

The following section recounts some events related to SASSER. Please refer to the timeline on the next page for other helpful information. This particular timeline covers significant events relevant to the study of the whole SASSER family event, starting from the discovery of the LSASS vulnerability and the "War of the Worms" to the discovery of other malware directly related to the said malware family. It attempts to establish the whole context of the SASSER event as discussed in this paper.

¹ Taken from CNN.COM article "New sasser version may be circulating." May 10, 2004.
<http://www.cnn.com/TECH/internet/archive/>.

**FIGURE 1:
Sasser Event
Timeline**



I. THE SASSER STORY

The War of the Worms

The year 2004 saw the advent of what can be described as the “War of the Worms.” WORM_MYDOOM was discovered on January 26, 2004 and TrendLabs consequently declared a medium risk virus alert to contain it. This highly publicized worm propagated via email and KaZaa shared folders. WORM_BAGLE, another family of network propagating mass-mailers, has been churning out variants since it was discovered on January 18, 2004. However, it did not get much public attention at first.

The first WORM_NETSKY variant was discovered on February 16, 2004. Like BAGLE, it propagated heavily via email and peer-to-peer (P2P) networks. However, the second NETSKY variant, released two days after the first, contained a particular routine that may have started the “War of the Worms.” This particular variant deleted registry entries employed by several MYDOOM variants. WORM_NETSKY.C finally declared an open war with the other malware families with this message:

```
<-<- we are the skynet - you can't  
hide yourself! - we kill malware  
writers (they have no chance!) -  
[LaMeRz->]MyDoom.F is a thief of  
our idea! - -  
< SkyNet AV vs. Malware >- ->>
```

WORM_NETSKY.D started deleting registry entries employed by BAGLE when it was discovered on March 1, 2004. PE_BAGLE.N, discovered on March 13, 2004, began its family’s anti-NETSKY campaign by deleting registry entries employed by NETSKY. Although MYDOOM variants did not delete registry entries employed by other malware they had this to say to the author(s) of NETSKY:

```
to netsky's creator(s): imho, skynet  
is a decentralized peer-to-peer neural  
network. we have seen P2P in Slapper  
in Sinit only. they may be called  
skynets, but not your shitty app.
```

The intense rivalry between NETSKY and BAGLE started around this particular variant’s release, and both families started spawning more variants in mutual retaliation. New variants of NETSKY and BAGLE were coming out so fast that at one point new variants from both malware families were released every day, or even twice in a single day. This was the state of things when SASSER had its debut on the Internet.

The Netsky-Sasser Connection

A link between the NETSKY family and the SASSER family was uncovered with the timely discovery of WORM_NETSKY.AC on May 2, 2004. Embedded text strings in the malware code declared that Skynet, the group that allegedly released the 29 notorious NETSKY variants in the wild, claimed responsibility for the SASSER variants.

Here is the actual message embedded in the malware code:

```
Hey, av firms, do you know that we  
have programmed the sasser  
virus?!?. Yeah thats true! Why do  
you have named it sasser? A Tip:  
Compare the FTP-Server code with  
the one from Skynet.V!!! LooL! We  
are the Skynet...
```

Also included is this snippet of SASSER source code allegedly serving as Skynet's proof of authorship:

```
Here is an part of the sasser sourcecode you  
named so, lol  
void TryLsass(char *pszIP){  
char arOS[130];  
if(detect(pszIP,arOS)==1)  
<rest of the code blocked>
```

More information on the NETSKY-SASSER connection can be seen in Section V.

Finding Sven

In the antivirus industry, the study of malware author profiles is not actually given much consideration. Adopting an "author-is-dead" perspective, antivirus engineers dissect and analyze every released malware without actual regard for the person who wrote it. Hence, like any piece of written work, each malware case is initially evaluated, processed and solved purely according to its own qualities.

However, in the case of this study, an entire section is devoted to the related news surrounding the suspected author of the SASSER worm. This is done for the sole purpose of completing the overall context of the whole SASSER family event.

**Figure 2: Sven
Jaschan, alleged
Sasser author**



On May 8, 2004, news articles around the world recounted the confession of an 18-year old German high school student who goes by the name Sven Jaschan. With the combined efforts of the Northwest Cyber Crime Task Force (a joint effort by the Federal Bureau of Investigation and Secret Service), German authorities and Microsoft, the teenager was tracked down in his home located in the small town of Waffensen (population: 920) in the western part of Lower Saxony, Germany.

The search for the said malware author gained momentum when a link between the NETSKY family and the SASSER family was uncovered with the timely release of WORM_NETSKY.AC, as discussed in the previous section. Another major lead in the investigation was the fact that the NETSKY variants X, Y, and Z included a routine that enables the launching of denial-of-service (DoS) attacks against www.nibis.de, among a few others (all of which are education sites). It was discovered that this particular Web site is the education server of Lower Saxony.

However, probably the most significant factor that led to Jaschan's arrest was the fact that Microsoft considered offering a reward of \$250,000 for information that would lead to the arrest and conviction of those responsible for the release of SASSER into the wild. Two days prior to Jaschan's confession, several informants contacted Microsoft offering information on the malware author. They backed up their tip by providing part of the SASSER worm code. Microsoft data protection official Sascha Hanke said that his company could say with great certainty that the informants obtained the said source code from Jaschan.

Aside from Jaschan's detailed testimony of the viruses he put out, he was clearly identified as the author of the SASSER worm because the source code was found in his computers, which were all confiscated. After questioning, he was released with pending charges, without having to pay bail. Prosecutor Detlev Dyballa said in reports that a trial could begin at the end of June. Criminal Office spokesman Detlef Ehrike said that the whole prosecution process could take quite a long time since officials still have to prepare hundreds of pages' worth of computer data for a possible court case. Jaschan was charged with computer sabotage, a crime that carries a maximum penalty of 5 years in prison in Germany. However, the fact that Jaschan was still a minor when he released the worms may significantly influence future court proceedings (he did not turn 18 until April).

Motives and Mind Games

Jaschan told officials that his original intention in creating the NETSKY variants was to remove viruses like MYDOOM and BAGLE from infected systems. This was discussed in one of the previous sections. Jaschan said that in the process of creating the NETSKY variants he developed SASSER. Jaschan explained to authorities that he released WORM_SASSER.E moments prior to his arrest to limit the damage caused by the other SASSER variants. That particular variant displayed the following text strings in a message box:

Figure 3:
Additional
message box for
SASSER.E



However, many speculate that Jaschan wrote SASSER to challenge other malware authors and to gain popularity as a highly skilled computer programmer in underground Internet communities. Some even speculate that he wrote SASSER to drum up business for his stepmother's PC-Help, a small computer store in Waffensen.

Since it is highly probable that Jaschan did not really operate alone, several people were rounded up for questioning after his arrest, including some of the informants. As of May 12, 2004, German authorities have already searched five homes near Jaschan's residence. Two have already admitted to receiving NETSKY source code from Jaschan, but only one had admitted to distributing it.

Microsoft and Internet security groups believe that Jaschan's arrest is a highly significant event in the history of information security. They think that the arrest will force malware authors to think twice before releasing potentially harmful worms. To them, his arrest confirms that offering rewards really does work, and that it is an effective way of opening up underground communities of malware authors.

In the meantime, one can only speculate that Sven Jaschan's arrest is the main reason why there are no new SASSER and NETSKY (and even BAGLE) variants currently in the wild. However, this is still not enough reason for computer users and IT administrators to drop their guards because some see this temporary malware lull as some sort of "eye in the storm" and that SASSER could just very well be the tip of an iceberg.

Friends and Foes

While Jaschan earned the ire of thousands of computer users all over the world, some people have already declared their support for him. Despite the damage done to millions of computers and thousands of networks, one leading German newspaper said in a commentary that there was a strange sense of national pride that a German student had outwitted the world's best computer experts.

Die Welt, a major German newspaper wrote: "Many of the (German) journalists who travelled to the province could not help but harbor clandestine admiration for the effectiveness of the worm." A new Web site, <http://support-sasser.homepage.dk/>, was even dedicated to raising money for the 18-year old Jaschan.

The Web site described Sasser as a "harmless wake up call." It describes Jaschan as some sort of scapegoat for Microsoft's failings, and that the alleged Sasser author "did the right thing by making this alarm call." It declared that serious criminals and/or terrorists could have deliberately written a destructive worm that exploits the same Microsoft vulnerability used by SASSER. However, as of this writing, the said support Web site is still closed and this message appears when attempting to access:

We're closed.

Actually, we were unable to get a hold of mr. Jaschan in a timely manner, so we have decided to stop our fundraiser. Losing the paypal account didn't exactly improve the case either, and it seems all other online payment services have even worse fees. All donations will of course be refunded to the extent that paypal permits usage of the locked account.

Cheers and much <3 from the support sasser team.

The Chaos Computer Club (CCC), one of the most influential hacker organizations in Europe, venerated for their well-intentioned hacks, offered indirect support to Jaschan by saying that Microsoft, too, should be held liable for the security holes in its operating systems that make them so vulnerable to worms and viruses.

The CCC became world famous when the group hacked the German "Bildschirmtext" (a Minitel-/Videotel-like system) and succeeded in getting a local bank to pay DM 134,000 into their bank account. The money was returned the next day in front of the press.

The Sasser worm and of course, Jaschan, already received the fickle attention of the media. This "popularity" is one factor that has made SASSER a considerable parallel of the highly effective WORM_MSBLAST.

"The rights have already been sold!" a man who opened the door of the family's detached home in the western Lower Saxony town of Waffensen told a Reuters reporter. He declined any further comment and closed the door, saying: "Goodbye."

While Jaschan wallows in the limelight, he may face possible damage claims amounting to millions of dollars from SASSER victims. These include U.S. carrier Delta Airlines, Australia's Westpac Bank, Goldman Sachs, and the British Coast Guard, among others.²

² Taken from Reuters.COM "Sasser originator may have been helping mum". May 10, <http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5093665>.

1
SASSER generates random IP addresses and verifies the availability of machines to infect by sending normal SMB packets to the random IPs at port 445. Meanwhile, SASSER also runs an FTP server on port 5554 (A,B,C,D,E) or 1023 (F) to facilitate its propagation routine.

2
SASSER sends a different exploit packet per platform to systems running Windows 2000 or XP.

2-a
This causes a buffer overrun on vulnerable systems that leads to the execution of a remote shell contained in the same exploit packet. As it runs, the remote shell uses port 9996 (A,B,C,F), 9995 (D), or 1022 (E) to listen for commands from remote systems.

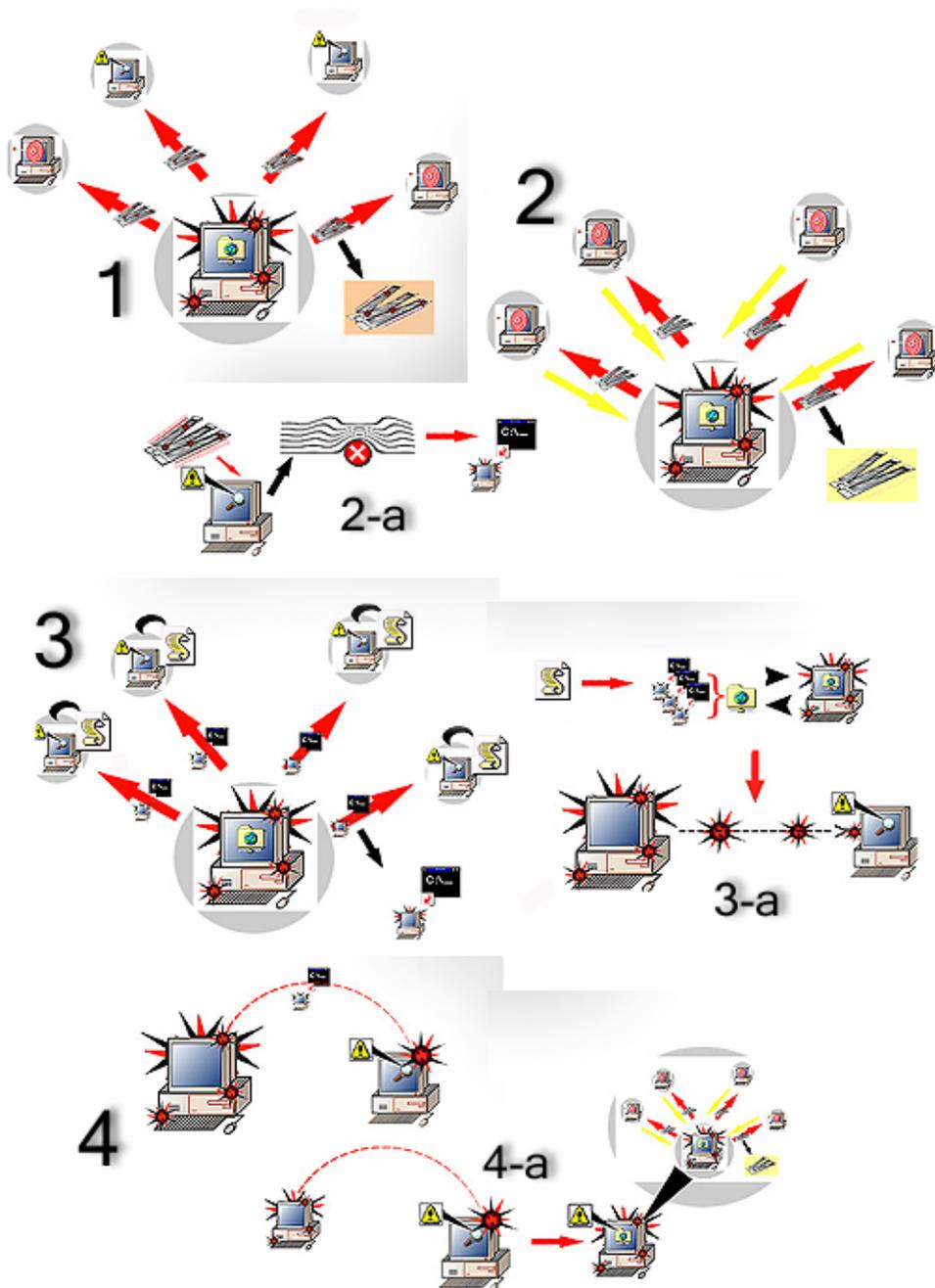
3
After sending the exploit packet, SASSER sends commands via the listening port to the remote shell running on receiving systems. The commands, executed by the remote shell, create, execute, and delete an FTP script file CMD.FTP.

3-a
The script file is a series of FTP commands that engage a session with the worm FTP server running on the infected machine. After the session is opened, succeeding commands on the script file download a copy of SASSER from the infected machine into receiving systems.

4
The SASSER copy sends a command to the remote shell running on the receiving systems. The command results in the execution of the downloaded SASSER copy.

4-a
Now the receiving systems are infected with SASSER to complete the propagation cycle. Newly infected systems perpetuate the worm's spread.

II. THE SASSER INFECTION ROUTINE



III. MSBLASTER and SASSER: Cut from the same cloth?

It took only **26** days from the announcement of the RPC DCOM vulnerability in Windows to the release of last year's first MSBLASTER worm. As seen in Table 1 below, SASSER took significantly less time, with just a **17-day** window between the LSASS vulnerability announcement and reports of user infection.³

**Table 3:
Sasser and
MSBlaster
Timelines**

SASSER Timeline			MSBLAST Timeline		
Date	Day	Event	Date	Day	Event
8-Oct-2003		Eeye Digital Security discovers and reports the LSASS vulnerability to Microsoft	July 2003		The Last Stage of Delirium discovers and reports RPC DCOM vulnerability to Microsoft
13-Apr-2004	1	Microsoft releases MS04-011 (cumulative patch covering several vulnerabilities including the LSASS vulnerability)	16-Jul-2003	1	Microsoft releases MS03-026 security bulletin (Buffer Overrun In RPC Interface Could Allow Code Execution)
14-Apr-2004	2	Immunity publicly announces/claims the release of LSASS and ASN.1 exploit codes to the public (functionality not known)	16-Jul-2003	1	LSD (Poland) releases exploit code after non-disclosure agreement expires
14-Apr-04	2	K-otik, a French Web site, posts the LSASS exploit on the Web	20-Jul-2003	5	Initial non-functional proof-of-concept code appears
29-Apr-2004	16	Public exploit that was confirmed to work and used by SASSER was released by "houseofdabus" according to Eeye	25-Jul-2003	10	Working exploit published by X-focus team (China)
30-Apr-2004	17	SASSER worm discovered; NAI, Fsecure, Sophos declares alert	25-Jul-2003	10	Metasploit (US) refines code to give remote command shell with escalated privileges on multiple versions of Windows
			26-Jul-2003	11	Ready to run version published
			31-Jul-2003	16	Concurrent hostile attacks occur at Stanford, UC Berkeley and MIT affecting more than 4000 computers
			11-Aug-2003	26	MSBlaster (Lovesan) appears with unaided, self-replicating exploitation of vulnerable hosts
			13-Aug-2003	29	Blaster hits Federal Reserve Bank of Atlanta, Maryland DMV and German automaker BMW
			14-Aug-2003	30	RpcSpybot variant uses same exploit but creates a backdoor that gives attacker control over PC using an IRC connection
			15-Aug-2003	31	Scandinavian bank closes all 70 branches, 440 servers infected
			18-Aug-2003	34	Good Worm variant finds infected computers, deletes Blaster and applies a patch (a.k.a Welchia or Nachi)

³ Significant portions on the MSBLASTER timeline were taken from http://farm9.com/pdf/CyberCrime_Timeline.pdf.

Perhaps it was the echoes of MSBLASTER's massive proliferation that drew public attention to the SASSER worm. After all, both are clever self-executing worms with alarming proliferation potential. Both are extremely destructive, not in terms of malicious payload, but in the aggressiveness of their propagation routine. And, both infiltrate Windows NT-based systems by exploiting known platform vulnerabilities.

The following subsections discuss the factors that make the self-executing SASSER worm so noticeably similar to MSBLASTER, as well as certain differences that mark SASSER as the faster, but more passive counterpart.

Basic Exploit Similarities

Both SASSER and MSBLASTER are essentially worms, meaning these malware types are self-contained programs that use malicious code to spread functional copies of themselves or their segments to other computer systems. Typically, the propagation takes place via network connections or through email attachments.

Your standard worm would usually require human intervention – such as opening an email – in order to be launched. A notable characteristic that differentiates SASSER and MSBLASTER from the common worm, on the other hand, is they take off on their own. No email attachments, no URL links. The possibility of infection becomes immediate simply by being a part of a network, such as the Internet or a Local Area Network (LAN), and by having an unpatched operating system. Another significant characteristic these two worms mirror is that they affect only Windows 2000 and XP systems.

	SASSER	MSBLAST
Date	1-May-03	11-Aug-03
Exploit	LSASS	RPC DCOM
Exploit packet sending	sends normal SMB packet first	directly sends exploit
Exploit port	445	135
Remote shell listening port	999,699,951,022	4444
FTP port	55,541,023	69
Payloads	SASSER.E displays message	Attacks Windows Update site
Attack symptoms	LSASS service crashes	RPC service stops
	Windows shuts down	Windows shuts down

RPC DCOM Exploit

The MSBLAST network virus exploits the buffer overflow in RPC DCOM. Windows-NT based systems, such as Windows 2000 and XP, use RPC (Remote Procedure Call) as the protocol used by a program to request services from another program on a server computer. DCOM (which stands for Distributed Component Object Model) is a protocol that enables programs to communicate over the network.

In Windows 2000 and XP environments, RPC DCOM is what enables separate components, such as clients and servers, to transparently send and receive information between COM ports on the same network. When the buffer in RPC DCOM overflows, an unauthorized user with local system privileges is able to execute any code on a target machine within the network.

Windows 95, 98, and ME systems communicate across the network using the NetBEUI protocol, instead of the RPC DCOM; hence these systems are not affected by MSBLASTER worm.

LSASS Exploit

A feature that makes the newer Windows systems more secure compared to older flavors such as Windows 95, 98, and ME is the user authentication requirement. To perform system changes, (i.e. installing or uninstalling a program), a user must have administrator privileges.

LSASS, which stands for Local Security Authority Subsystem Service, and is in charge of Windows 2000 and XP security mechanisms. It is the component that verifies the validity of users logging on to the computer and generates the process responsible for authenticating users for the Winlogon service.

If authentication is successful, LSASS generates the user's access token. A user identified as Administrator can perform changes to the computer system, while a user without administrator rights cannot. When SASSER exploits the LSASS vulnerability and causes a buffer overflow, a remote malicious user, is able to perform applications using administrator privileges over the network.

Windows 95, 98, and ME do not have the LSASS component, and need not authenticate administrator privileges to run processes in the system. Hence, the SASSER worm does not affect these systems.

Infection Technique

Earlier we mentioned how SASSER and MSBLASTER are able to perform their infection routines automatically and do away with user intervention (opening an email attachment or clicking on a malicious URL). The technique uses the malware's basic exploits to perform mass propagation routines across networks. MSBLASTER uses port 135 to find vulnerable systems to infect – a port used by the DCOM protocol that is open by default.

Once the target machine is accessed, the worm opens the target machine's port 4444 to run a remote shell. The remote shell simulates an FTP server and downloads a copy of the MSBLASTER worm via port 69.

Similarly, SASSER uses port 445 to scan for vulnerable systems – a Server Message Block (SMB) port used for carrying out the LSASS protocol in Windows file and print sharing, as well as numerous other network services. Once the target machine is accessed, SASSER opens ports 1022, 9995, or 9996, depending on the variant, to run a remote shell. The remote shell then simulates an FTP server and downloads a copy of the SASSER worm via ports 1023 (for WORM_SASSER.E) or 5554 (for all other variants).

Installing updates MS03-026 and MS04-011 from the Microsoft Web site should effectively patch the RPC DCOM and LSASS vulnerabilities in Windows and prevent the insidious activities of these two worms. However, an unpatched system in a networked environment is fair game, risking automatic infection. Because the worms are self-executing, the user remains unaware of the malware presence until the computer begins displaying classic infection symptoms.

Symptoms

Figure 4:
System Shutdown
Box: the most
visible symptom of
an MSBLASTER
worm infection.



Computers infected by both SASSER and MSBLASTER may shut down or restart if they receive an exploit code that causes the LSASS process (for SASSER) or the RPC DCOM process (for MSBLASTER) to crash. In the case of MSBLASTER, the worm does not necessarily cause the shutdown process directly. Instead, the worm creates threads that generate random IP address numbers.

Packets are sent to these random IP addresses to find RPC DCOM vulnerabilities. In addition to random IP address scanning, MSBLASTER also launches an aggressive denial-of-service attack against the windowsupdate.com website, where 40-byte packets are sent to windowsupdate.com at 20 millisecond intervals.

All that activity causes the Remote Procedure Call service in Windows 2000 and XP to stop. And this, in turn, causes NTAUTHORITY\SYSTEM to reboot.

The computer displays a “shutdown” message box. And at the end of the 60-second countdown, the system reboots. This disruptive reboot sequence repeats each time the computer goes online.

The effect differs with Windows 2000 systems. Similar to the process described above, the RPC process also stops, but it does not cause NTAUTHORITY\SYSTEM to reboot automatically. Since many services depend on RPC, some services may not function properly.

Systems infected by the SASSER worm do not have denial-of-service attack routines. However, the worm also creates threads that generate random IP addresses. Packets are sent to these IP addresses to scan for unpatched systems that are vulnerable to the LSASS exploit. Depending on the SASSER variant, packets sent to these random IP addresses range from 512 to 40,960 packets per second.

Due to such massive packet transmission, LSASS.EXE crashes and an error message appears on screen. Similar to the MSBLASTER infection, a shutdown screen appears, and the system reboots at the end of 60 seconds.

Figure 5:
This error message
appears after
LSASS.EXE crashes.

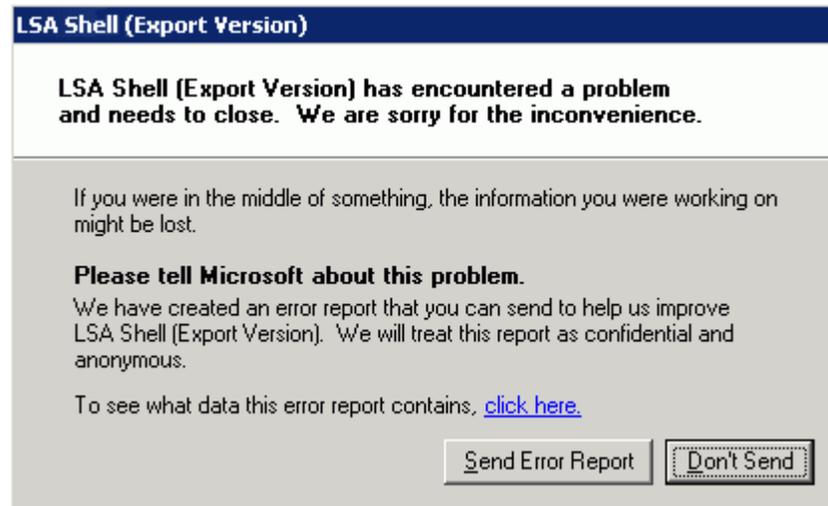
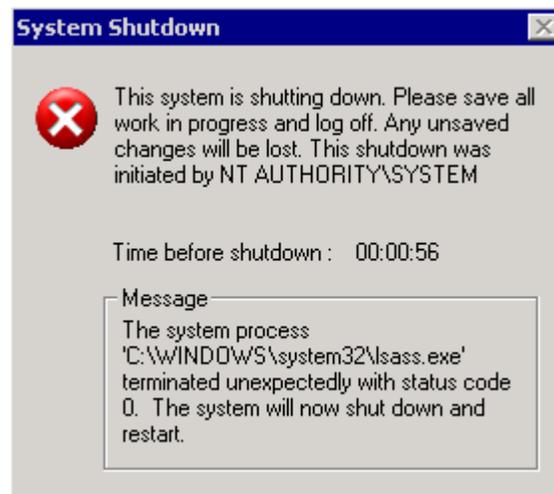


Figure 6:
This message
appears warning
of Windows
system shutdown
with automatic
restart in 60
seconds.



The preceding message appears warning of a Windows system shutdown with automatic restart in 60 seconds. Since both files also drop copies of themselves in the Windows system directory, a less visible indication of infection can be seen in the files outlined in the following table.

**Table 3:
Dropped files of
SASSER and
MSBLASTER**

MALWARE	FILE NAME
MSBLAST	Msblast.exe
	TFTP* files – result of a failed download routine
WORM_SASSER.A	Avserv.exe
WORM_SASSER.B	Avserv2.exe
WORM_SASSER.C	Avserv2.exe
WORM_SASSER.D	Skynetave.exe
WORM_SASSER.E	Lsass.exe
WORM_SASSER.F	Napatch.exe

Effects on Users

Disruptive enough as the system reboot payload may be, these worms also use up exhaustive amounts of system resources. Computers infected by SASSER and MSBLASTER notice dramatic decrease in processing speed, even while performing the simplest of applications.

Additionally, continuous packet transmission to random IP addresses may also cause network congestion. This in turn could adversely affect network applications such as file and print sharing.

SASSER: Fast, Not Sassy

Unlike the MSBLASTER worm, SASSER neither performs denial-of-service attacks against any given Web site, nor any other malicious functions. SASSER's primary function is to propagate, and it does little else outside worming its way into the world's vulnerable Windows NT-based machines. Next to MSBLASTER's aggressive denial-of-service attack on the Windows update Web site, SASSER presents a lethargic comparison.

However, SASSER saw a speedier release compared to MSBLASTER, which started spreading 26 days after the RPC DCOM vulnerability patch release. It may have been the MSBLASTER precedent that fueled active media coverage, forewarning users of the SASSER spread and advising basic preventive steps. Yet despite all the hoopla, SASSER has accomplished its expansive proliferation function.

To date, Trend Micro World Virus Tracking Center reports **56,000** total infections attributed to the most prolific SASSER variant, WORM_SASSER E. This surpasses the **52,000**-infection mark of MSBLASTER.C. Note that MSBLASTER.C was discovered on August 13, 2003, and that WORM_SASSER.E is only approximately a month old as of this writing.

IV. SASSER Variant Comparison

**Table 4:
Summary of Sasser
Variants**

	A	B	C	D	E	F
Date of Discovery	April 30, '04	May 1, '04	May 2, '04	May 3, '04	May 8, '04	May 10, '04
Size (in Bytes):	15,872	15,872	15,872	16,834	15,872	74,752
Platform	Execution	2003	2003	2003	2003	2003
	Execution and propagation	2000, XP	2000, XP	2000, XP	2000, XP	2000, XP
Exploit	04-011	04-011	04-011	04-011	04-011	04-011
Autostart registry key	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	HKLM\Software\Microsoft\Windows\CurrentVersion\Run
Autostart registry entry name	avserve.exe = %Windows%\avserve.exe	avserve2.exe = %Windows%\avserve2.exe	avserve2.exe = %Windows%\avserve2.exe	skynetave.exe = %Windows%\skynetave.exe	avserve.exe = %Windows%\sassets.exe	napatch.exe = %Windows%\napatch.exe
Compression	PE Compact v.2	PE Compact v.2	PE Compact v.2	PE Compact v.2		
Compilation					.NET	
Dropped file name	AVSERVE.EXE	AVSERVE2.EXE	AVSERVE2.EXE	SKYNETAVE.EXE	LSASSS.EXE	NAPATCH.EXE
Mutex Name	single execution	Jobaka3l	JumpallsNlStilt	JumpallsNlStilt	SkynetSasser Version With PingFast	Skynetnotice
	use unknown		Jobaka3	Jobaka3	Jobaka3	billgate
Ports used	to initiate buffer overflow	445	445	445	445	445
	to listen for commands	9996	9996	9996	9995	1022
	to accept FTP command	5554	5554	5554	5554	1023
No. of threads created	128	128	1024	128	128	128
No. of attacks per second	512	5120	40960	5120	5120	512
No. of infections at peak	< 11,000	~ 15,000	~ 14,000	> 4,000	~ 25,000	1
No. of infections to date*	20,240	29,466	22,345	9,212	56,164	1
Log file	WIN.LOG	WIN2.LOG	WIN2.LOG	WIN2.LOG	FTPLOG.TXT	WIN.LOG
Notable details				improved scanning routine by pinging target machine to verify Internet connectivity	improved scanning routine by pinging target machine to verify Internet connectivity	copycat re-release of Sasser A
				does not seem to run on Win2K due to this routine	does not seem to run on Win2K due to this routine	
					deletes BAGLE variants	
					Displays a window, reminding users to download patch	

WORM_SASSER.A

The first SASSER variant, WORM_SASSER.A, was discovered on May 1, 2004. It was the first malware discovered to exploit the LSASS vulnerability, a vulnerability that was discovered about five months before. (For the complete SASSER timeline, please refer to Figure 1).

While a patch for this vulnerability had been available since April 13, 2004 in the Microsoft Web site, user awareness was minimal regarding the LSASS vulnerability when WORM_SASSER.A was released. Hence, WORM_SASSER.A was able to infect 20,472 machines to date, according to the Trend Micro World Virus Tracking Center.

WORM_SASSER.A creates only one mutex, *Jobaka3l*. It also looks for this mutex in target machines to prevent re-infection. It drops the file AVSERVE.EXE in the Windows folder, and creates a corresponding registry entry to ensure that this worm executes at every Windows startup. It generates 128 target IP addresses every 0.25 seconds, resulting in 512 attacks per second. This is reflected in Table 4 as "No. of threads created" and "No. of attacks per second."

This variant exhibits the typical propagation routine for SASSER, as described in Section II. It uses port 445 to initiate the buffer overflow of a target machine. Once this overflow takes place, port 9996 is used to listen for commands from the infected machine. When the infected machine receives the signal that the target machine is listening, it commands the target machine to open a command shell to download a copy of the worm. The target machine opens its port 5554 to download the worm copy.

(Note: Port 445 is the most commonly used port to share files over the network. This port is reserved for Windows OS usage. Ports 9996 and 5554 are ports that can be used by ordinary user processes.)

WORM_SASSER.B

WORM_SASSER.B was discovered on May 1, 2004. It drops the file AVSERVE2.EXE, establishing a connection with WORM_SASSER.A (which drops the file AVSERVE.EXE). A registry entry is also created for automatic execution at every system startup.

This variant creates two mutexes, as compared to WORM_SASSER.A which only created one. The mutex that WORM_SASSER.B creates to avoid re-infection of a system is *JumpallsNlsTillt*. It also creates the mutex *Jobaka3* (again with reference to mutex *Jobaka3l* of WORM_SASSER.A), but this particular mutex doesn't seem to have any particular purpose.

Compared to the first variant, which had an infection frequency of 0.25 seconds, WORM_SASSER.B created target IP addresses ten times faster, generating 128 threads every 0.025 seconds. This results in 5,120 attacks per second, ten times more than WORM_SASSER.A.

The increase in attack rate possibly caused the number of infections of WORM_SASSER.B to be slightly higher than WORM_SASSER.A. Based on WTC statistics, the B variant has infected around 30,000 computers, as of this writing.

WORM_SASSER.B uses the same ports as WORM_SASSER.A to cause the LSASS buffer overflow (port 445), to listen for commands from the infected machine (port 9996), and to download the worm copy (port 5554).

WORM_SASSER.C

This SASSER variant was discovered on May 2, 2004, and has the same dropped file name and mutex names as WORM_SASSER.B. Its only difference from WORM_SASSER.B is the number of threads it creates, and thus the number of attacks it is able to generate per second.

WORM_SASSER.C creates 1024 threads per second, almost a tenfold increase in thread number compared to WORM_SASSER.A and WORM_SASSER.B. It has the same frequency of infection as WORM_SASSER.B, 0.025 seconds, thus yielding an attack rate of 40,960 attacks per second.

Despite this drastic increase in attack rate, the number of successful infections of WORM_SASSER.C is less than those for WORM_SASSER.B. WORM_SASSER.C has infected approximately 22,000 computers to date (WTC data). This number, while not insignificant, is considerably less than what can be expected from a variant with such a huge increase in attack rate. This is probably due to increased user awareness, as the first two SASSER variants caused virus alerts to be declared by most antivirus companies, and were the subject of intense media hype.

WORM_SASSER.C uses the same ports as the A and B variants to cause the LSASS buffer overflow (port 445), to listen for commands from the infected machine (port 9996), and to download the worm copy (port 5554).

WORM_SASSER.D

The fourth SASSER variant was discovered the day after WORM_SASSER.C. It drops the file SKYNETAVE.EXE, and also modifies the registry so that it executes at every Windows startup. It also has two mutexes: the seemingly purpose-less *Jobaka3*, and the mutex *SkynetSASSERVerionWithPingFast* that it creates to avoid system re-infection. It has the same attack rate and number of created threads as WORM_SASSER.B.

It is possible that future malware will look for vulnerable systems by searching for the mutex *Jobaka3*. This may provide malware authors with a pool of possibly vulnerable machines more in number than those affected by any one SASSER variant, since this mutex is created by SASSER variants B, C and D. This may be a precursor to a multistage malware or to second wave attacks by a malware yet to be perfected by malware authors.

WORM_SASSER.D is also the first SASSER variant to make an allusion to the connection between the SASSER family and the Netsky family, as seen from its second mutex name. More information on the battle between worm families can be found in Section I. As advertised by its second mutex name, WORM_SASSER.D has a modification in its code, which enables it to speed up its scanning routine. It sends out an ICMP echo request to its target machine before attempting to make a connection, a feature not present in all the other variants. In short, it “pings” its target machine to make sure that the target machine is indeed connected to the Internet. Once the ICMP request

is confirmed, it is only then that the infected machine connects to the target. This speeds up the infection rate of WORM_SASSER.D, at what might have been an attempt by the author to create a more pervasive malware as compared to the first three variants.

However, despite this code improvement, this variant does not seem to run properly on some Windows 2000 systems. It can infect, but cannot propagate from a machine running on Windows 2000 because its ICMP echo request routine uses an import symbol from the dynamic link library IPHLPAPI.DLL (which does not exist in Windows 2000). WORM_SASSER.D uses ports 445 and 5554 for the same functions as all previous variants, but uses port 9995 to listen for commands from the remote infected machine. Port 9995 is also an ordinary port, similar to 9996.

WORM_SASSER.E

WORM_SASSER.E, released five days after WORM_SASSER.D, and hours before the arrest of the alleged author, remains the most prolific of the SASSER variants. This variant has infected around 56,000 computers worldwide according to WTC. These numbers are surprising, considering that after four variants users should already have patched their systems, and that WORM_SASSER.E possesses the same malfunction in Windows 2000 as WORM_SASSER.D.

It is possible that the malware author or authors were aware of the inconsistency regarding WORM_SASSER.D and WORM_SASSER.E in Windows 2000, since there is a five-day gap between their releases. The fact that it was released by the alleged author hours before his arrest opens the possibility that despite the Windows 2000 bug known to be present in WORM_SASSER.D, Jaschan went ahead and released E as a “parting shot” before he was captured by police. The presence of this code malfunction in Windows 2000 further fuel claims by antivirus experts that the author of SASSER was not a very experienced programmer.

This variant drops the file LSASSS.EXE into the Windows folder and also executes at every Windows startup. It creates the mutex *Skynet/Notice* giving strength to the assertion that the same group of people created SASSER and Netsky. Another characteristic of WORM_SASSER.E compounding this claim is that it deletes processes connected to the Bagle family of worms, the self-declared “enemy” of the Netsky family.

WORM_SASSER.E uses port 445 for the buffer overflow, but uses port 1022 to listen for commands from the infected machine, and port 1023 to download the worm copy. Ports 1022 and 1023 are Windows reserved ports, meaning they don't have a specific function as of now in any Windows platforms but may have one in future Windows versions.

Another difference between the previous variants and WORM_SASSER.E is that WORM_SASSER.E displays an additional message box aside from the message box informing users that they are exposed to the LSASS vulnerability and that they must patch their systems immediately. This is discussed in Section I.

WORM_SASSER.F

The F variant of SASSER is a re-release of WORM_SASSER.A. It differs from the A variant only in the name of the dropped file and the mutex name, which were edited using a hex editor. WORM_SASSER.F also has a different file size than WORM_SASSER.A.

The author of this variant may have changed the size to avoid detection by patterns for WORM_SASSER.A. Otherwise, this variant has the same functions and behaviors as WORM_SASSER.A, which can only mean that it was created by a copycat — someone out to claim notoriety for having released a virus without having to do much coding.

A Final Word on the Sasser Variants

The study of the different variants of SASSER makes one wonder what the intentions of the malware author(s) were. Evidently, each variant didn't seem to be an improvement over previous one, with the exception of WORM_SASSER.E. In terms of SASSER evolution, there isn't much of an evolution to speak of, as compared, say, to the Bagle family, whose variants seem to be an enhancement of previous versions.

Aside from the differences in dropped file names and mutex names, there was little else that changed with each release. Most malware authors make it a point to change the file names and mutex names of malware variants in order to avoid detection by already-released patterns. The number of attacks per second increased from 512 to 40,960 in variants A to C but reverted back to 512 with variants D and E. The ports used were more or less the same across all variants, except for E, which used Microsoft reserved ports possibly because the author believed it could infect networks that had failed to block these ports.

V. NETSKY and SASSER: Brothers in crime?

The move from NETSKY to SASSER (if indeed only one person or group is responsible for both) says much about the lifespan of vulnerability-based malware as compared to mass-mailers that rely on social engineering. Perhaps the most important point to consider is that vulnerability-based malware eventually die out after the computing world patches up. For example, WORM_NETSKY.P (discovered on March 22, 2004) continues to sit on the top spot in Trend Micro's list of Top Threats while the world waits for the eventual demise of SASSER. Before moving on, it's better stated that prevalence doesn't necessarily translate to the destructive potential of a malware.

WORM_SQLP1434.A, popularly known as Slammer (discovered on January 24, 2003), suffered an early death. However, even if it was designed to live a short life, it is still undeniably a significant malware in terms of damage caused. This is to say that SASSER is by no means less significant than WORM_NETSKY.P.

Both NETSKY and SASSER share a common root according Sven Jaschan's accusers and published technical documentation on the NETSKY and the SASSER variants. However, the transition from socially engineered mass-mailers to vulnerability-based creations, despite the fact that the earlier creation outlived and *out-thrived* the latter, should not distract one from seeing the pattern (or the distinct absence of it) that connects NETSKY to SASSER.

The Antivirus Virus

On its first variant, (WORM_NETSKY.B), the NETSKY creator(s) jumped into what looked like a free-for-all worm battle concerning certain MYDOOM variants. This is discussed in the subsection entitled "War of the Worms."

While this went on, NETSKY author(s), apparently suffering from a bad case of identity crisis, kept insinuating that they were the “good” virus writers. They practically described NETSKY as an *anti-piracy* and *anti-virus* virus. WORM_NETSKY.Q (discovered on March 28, 2004) attacked P2P networks and cracked/serial number sites to support this assertion. Body text on WORM_NETSKY.R (discovered on March 31, 2004) criticized backdoor routines in certain BAGLE variants.

Notice, however, that after four days, a new variant, WORM_NETSKY.S (discovered April 4, 2004), contained its own backdoor routines. Body text on WORM_NETSKY.S maintained that the backdoor code only supported propagation. The routine, however, practically opened up a host of possibilities for the remote malicious user.

Backing up theories that NETSKY and SASSER had the same creator(s), WORM_SASSER.E (discovered on May 8, 2004) exhibited the same *good guy-bad guy* identity confusion by displaying a message that warned infected users of the perils of not patching against the LSASS vulnerability. The author apparently left out the fact that SASSER is actually the worst LSASS peril. Sven Jaschan tried mitigating the charges against him by pointing out the message and his good intentions.

Will his lawyer mention that backdoor routines in NETSKY had absolutely no malicious purpose if he is proven to be responsible for both malware families?

SASSER Studies in NETSKY

The technical merits of SASSER and its obvious inspiration, MSBLASTER or the Blaster worm (discovered on August 11, 2003) is discussed in Section III. NETSKY is not a close routine relative of SASSER, but if theories of a common authorship are true, NETSKY should contain clues on the development of SASSER.

Comparatives of the NETSKY and SASSER variants (see previous section on SASSER variants for details) basically present authorship that is not concerned with releasing malware in stages. The BAGLE author has been shown to test certain features with each release, probably in an attempt to eventually come up with the ultimate worm.

NETSKY variants, on the other hand, seem to reveal that its author(s) has no clear desire to improve the worm. The same attitude can also be observed from the study of the SASSER variants. It appears that SASSER modifications have been done on impulse and, probably, out of curiosity. There are key points in the NETSKY release timeline (located in Appendix A) that would show this preference to experimentation.

On April 14, 2004, WORM_NETSKY.V was discovered to be quite a distinct NETSKY variant. NETSKY.V was not a mass-mailer in the strictest sense. Although it sent out email as part of its propagation routine, what it actually sent out was exploit code embedded in email.

It took advantage of Internet Explorer vulnerabilities MS03-032 and MS03-040 that had patches released August and October the year earlier. The WORM_NETSKY.V method, which is often likened to the SASSER propagation routine (refer to routine image), uses an exploit to cause vulnerable systems where its email is opened to download the worm from infected machines. Note that the use of infected machines as a download infection vector is relatively rare since most download vectors are fixed sites that are controlled by

malware authors. Therefore, WORM_NETSKY.V may also strengthen the idea of common authorship.

Jaschan's lack of patience to refine his code is common in young malware authors. Likewise, NETSKY appears to have been written by authors who resort to impulsive application and experimentation of routines. NETSKY also reveals no deeper motive other than to compete with other malware families.

VI. The SASSER Bandwagon

The SASSER worm has left in its wake a trail of wannabes — worm programs that similarly exploit the Windows LSASS flaw or minor derivatives freeloading on the LSASS bandwagon. In much the same way, several worm programs also exploited the RPC DCOM security hole (MS03-026) to propagate, after the infamous MSBLASTER worm brazenly got away with “successful” exploitation of the vulnerability. This section discusses the following worms that emerged after SASSER's deployment into the wild and capitalized on the same propagation mechanism:

- CYCLE
- DABBER
- KORGO

Although their fray into the wild is not as rapid and as widespread as SASSER's, these worms inject certain elements to the LSASS exploit that seem like bids to top the earlier worm.

CYCLE

The CYCLE worm, the next LSASS worm discovered after SASSER, appeared in the wild on May 10, 2004. It is the first known anti-SASSER worm program. Clearly riding in the LSASS bandwagon, this worm contains a long, politically tainted message about freedom in Iran.

On execution, it creates a copy of itself in the Windows folder using the file name of a legitimate Windows file, svchost.exe. It also creates the text file cyclone.txt in the same folder, which interestingly contains the said politically tainted message.

It creates the following mutexes, which are similar to the mutex objects created by SASSER worms:

- SkynetSASSERVersionWithPingFast
- Jobaka31
- JumpallsNlsTillt
- Jobaka3

It also terminates the following processes:

- msblast.exe
- avserve.exe
- avserve2.exe
- skynetave.exe

Note that these processes are associated with SASSER and NETSKY variants.

When the system date is May 18, it launches denial-of-service attacks against the following Web sites:

- www.irma.com
- www.bbcnews.com

Employing the basic propagation routine of SASSER, the CYCLE worm opens TCP port 3332 as infection marker. It accepts connections on this port and immediately closes them to signify that the host system is already infected.

It generates a random target IP address and attempts to connect to TCP port 445 (the port associated with the LSASS flaw).

It runs a TFTP server on UDP port 69, which sends a copy of the worm through that port. Then, it runs a remote shell, which downloads a copy of itself as cyclone.exe from the said TFTP server, and executes the copy. A TFTP client named "tftp" should be present in the path of the remote computer where the remote shell runs.

It also connects to TCP port 3332 on the remote system to check if the system is already infected. When the connection attempt succeeds, it assumes that the computer is already infected and ends the infection attempt.

DABBER

The DABBER worm (discovered May 13, 2004); 13 days after the first SASSER variant turned up, scans the network for SASSER infected systems and uses these systems as launching pad for its network propagation.

It scans random subnets for sequential IP addresses on port 5554, scanning the network for systems infected by SASSER worms. When it finds an infected system, the worm exploits the vulnerability in the FTP server component of the SASSER worm. It binds to a command shell to port 8967 and uses the shell to make the infected system download and execute the worm via FTP.

It also deletes the registry autostart values associated with NETSKY and SASSER variants to continue with what is known to be a protracted war between the malware authors.

KORGO

The KORGO worm is now on its 9th variant and is still aggressively replicating. Its first variant was discovered on May 22, 2004 — 22 days after the rapid spread of SASSER.A into the wild.

Like SASSER, it exploits the flaw in the Windows LSASS. It generates random IP addresses to attack and creates threads that exploit the LSASS flaw on TCP port 445, enabling a remote system to connect to the infected host and download the worm copy.

Although both worms basically exploit the LSASS vulnerability to spread, KORGO builds on the functionalities of the earlier worm. Unlike SASSER, each KORGO variant drops a randomly named copy of itself, making its presence more difficult for infected users to

detect. KORG0 also injects its process into Windows Explorer (EXPLORER.EXE) so that it cannot be detected in memory.

The KORG0 worm also has backdoor functionalities. It enables remote access via different TCP ports, virtually leaving the infected system open to access and manipulation. It also uses IRC to further enhance its backdoor capabilities, connecting to a list of IRC servers and channels, where backdoor commands can be issued and processed locally on the system.

Falling Off the Wagon

The SASSER outbreak has spurred a relentless drive to protect systems against malware programs that exploit the LSASS flaw and as such hampered the propagation of such worms as CYCLE, DABBER, and KORG0. These worms may never get to “enjoy” the rapid spread of SASSER, with most computers already patched and antivirus solutions deployed. To date, only KORG0 is actively replicating variants, with considerably limited infection counts.

VII. Implications of a SASSER

SASSER was released at a time when antivirus support services were recently bombarded by outbreaks due to the protracted BAGLE-NETSKY war. Malware activity in Q1 of 2004 surpassed total activity of all viruses combined the previous year (source: APAC Marketing, Trend Micro Inc.). The maturity of the AV industry was accelerated by the spate of virus outbreaks in Q1.

The sheer difference between the mass-mailers of Q1 and the vulnerability-based LSASS worm required a different reaction from the AV industry altogether. SASSER also managed to evade generic detections by even the most effective engines and proactive pattern makers. To contrast, different vendors detected the most prominent NETSKY and BAGLE variants generically with proactive patterns created using initial variants. Even the file infection capabilities of the infecting BAGLEs, PE_BAGLE.P and PE_BAGLE.Q, were detected with patterns designed to catch file infectors. The absence of proactive patterns practically allowed a later variant, SASSER.E, to surpass infection activity of the earlier variants despite the likelihood that a lot of patching has occurred (source: Trend Micro World Virus Tracking Center).

Another important note for SASSER is the fact that patching proved too slow. Despite the media activity for SASSER, KORG0, which was released almost a month after SASSER.A, still managed to make the rounds with the help of the LSASS security hole. Based on infection statistics for SASSER.F, which was released nine days after SASSER.A, most of the computer world should have had patched by that time. But this reaction time was still too slow - SASSER authors were able to come up with SASSER.A within a day (or a few hours) after *Houseofdabus* released the exploit code. It took the computing world almost a month after the Microsoft patch release on April 13, 2004 and 3 major outbreaks to patch up. It would also be good to note that several major AV vendors called out an alert for a KORG0 variant.

Apart from patching and pattern making, the SASSER incident also showed us a more direct way to counter an outbreak: arrest the author. The fingers pointing at Sven Jaschan seem to have the facts on their side. And it is quite unlikely that people will

question the arrest, since Jaschan has admitted authorship. Strangely, there are still some indications that SASSER and NETSKY variants, in the guise of other names and additional routines, will still plague us. Perhaps, a new group of kids inspired by Jaschan's *anti-virus virus* or "good-willed" virus authorship might pursue the same cause. Virus authorship has proven to be as easy as vandalizing silly hate messages against a rival gang.

The AV industry, however, has taken up the challenge of providing non-traditional solutions to the non-traditional misdemeanors of today, which cost millions of dollars in losses. AV scanners that can run at the network level can now preempt the traditional problem at the desktop level – the user. These scanners can detect known effective exploit packets, which are usually recycled by malware authors. This increases the chances of generically detecting malware, which is good in general. Detecting malware at the network before they can actually run on the desktop is an even better plus.

These network-based scanners, when implemented with vulnerability assessment, can be used to effectively isolate machines that are found vulnerable. This will foil most vulnerability-based malware as long as AV support services release corresponding exploit patterns fast – a challenge considering how soon SASSER was released after the posting of the exploit code it used.

SASSER, despite its implications and the impressive solutions from the antivirus industry, is not entirely new. It is practically an MSBLASTER rip-off. However, its use of a new vulnerability technically makes it new. It also shows how generic detections and the smartest patterns cannot always stop malware. These solutions play a game of chance with customer systems. The best solution still proves to be effective response, which includes product patch delivery and information provision. The next spate of outbreaks should demonstrate which vendor measures up.

Appendix A: Sources

I. News

http://www.dw-world.de/english/0,3367,1446_A_1201704_1_A,00.html
<http://www.net-lexikon.de/Chaos-Computer-Club.htm>
<http://www.securityfocus.com/news/8581>
<http://www.cnn.com/TECH/internet/archive/>
<http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5053982>
<http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5079960>
<http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5080505>
<http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5081009>
<http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5093665>
<http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5080447>
http://news.com.com/2100-7349_3-5203791.html
<http://www.reuters.co.uk/newsArticle.jhtml?type=topNews&storyID=5121081>

II. Virus Information and Statistics

<http://www.trendmicro.com/vinfo/>
<http://www.trendmicro.com/map/>

<http://www.trendmicro.com/NR/rdonlyres/8472BABE-B8AE-4DF9-806D-F48308D88BD2/9565/VirusRoundup.pdf>

http://farm9.com/pdf/CyberCrime_Timeline.pdf

Chen, Tracy. "Most Virus Alerts Ever Issued in Q1 2004 – 6.6 Times The Same Quarter Last Year". APAC Marketing Outbreak in-depth article, April 1, 2004.

Gordon, Jason. "Lessons from Virus Developers: The Beagle Worm History Through April 24, 2004."

III. General Information

<http://sbc.webopedia.com/TERM/R/RPC.html>
<http://sbc.webopedia.com/TERM/D/DCOM.html>
<http://www.iana.org/assignments/port-numbers>

Appendix B: Detailed Timeline

8-Oct-03	LSASS Vulnerability discovered	Eeye Digital Security discovers and reports the LSASS vulnerability to Microsoft
16-Feb-04	WORM_NETSKY.A	1st NETSKY variant
18-Feb-04	WORM_NETSKY.B	Netsky starts to attack other malware - terminates MYDOOM.A, MYDOOM.B, and MIMAIL.T
25-Feb-04	WORM_NETSKY.C	Changes anti-malware attack mode to registry deletion
1-Mar-04	WORM_NETSKY.D	Starts attacking BAGLE
13-Mar-04	PE_BAGLE.N	BAGLE starts fighting back against NETSKY
14-Mar-04	PE_BAGLE.P	Body text attacks begin
22-Mar-04	WORM_NETSKY.P	First exploit use - old commonly used MIME header exploit MS01-020
28-Mar-04	WORM_NETSKY.Q	DoS against P2P and crack sites - claims further that it is an antivirus and anti-piracy virus
31-Mar-04	WORM_NETSKY.R	Attacks BAGLE backdoor routines through body text, but does not delete registry entries
4-Apr-04	WORM_NETSKY.S	Backdoor use (body text maintains that backdoor can be used for propagation only)
6-Apr-04	WORM_NETSKY.T	DoS against P2P and crack sites (4/14-4/23)
7-Apr-04	WORM_NETSKY.U	Buggy release
13-Apr-04	LSASS Vulnerability Patch released	Microsoft posts MS04-011 advisory with cumulative LSASS patch
14-Apr-04	WORM_NETSKY.V	Infects by contacting IP after email is opened; Uses exploits MS03-032 and MS03-040; No attachment; Noted similarity to SASSER
15-Apr-04	WORM_NETSKY.W	Apparently modified from original writer's source; Restored anti-malware routines; Sends email to a chris_sexana@aol.com
20-Apr-04	WORM_NETSKY.X	First variant to perform DoS against education sites associated to Jaschan; No anti-malware routines
20-Apr-04	WORM_NETSKY.Y	Dares BAGLE in body text
26-Apr-04	WORM_NETSKY.AA	Surprisingly no anti-malware; No DoS
26-Apr-04	WORM_BAGLE.X	Anti-NETSKY registry deletion; Use of JPEG attachment
27-Apr-04	WORM_NETSKY.AB	Deletes BAGLE registry entries; Talks "revenge" in body text
29-Apr-04	WORM_BAGLE.Z	Anti-NETSKY registry deletion
29-Apr-04	LSASS Vulnerability Exploit Code	<i>Houseofdabus</i> releases exploit code utilized by SASSER
1-May-04	WORM_SASSER.A	SASSER.A is discovered; Several AV vendors declare outbreak; WTC total infections = 20,499
1-May-04	WORM_SASSER.B	SASSER.B tries to infect 10 times more machines per second; WTC total infections = 30,206
2-May-04	WORM_SASSER.C	SASSER.C tries to infect 40 times more machines per second than A; WTC total infections = 22,935

2-May-04	WORM_NETSKY.AC	Pretends to provide fix tool for SASSER.B, BLAST.B, MYDOOM.F, or NETSKY.AB; Deletes BAGLE registry entries
3-May-04	WORM_SASSER.D	SASSER.D tries to infect 10 times more machines per second than A (same as .B); WTC total infections = 9,434; Contains Win2k bug; Uses <i>skynet</i> for file name
8-May-04	WORM_SASSER.E	Deletes BAGLE registry entries; Displays message box with vulnerability warning; Contains the same .D bug; WTC infections = 56,900
8-May-04	Jaschan's arrest	18-year old Sven Jaschan is arrested in Waffensen (pop. 920), Lower Saxony, Germany
10-May-04	WORM_SASSER.F	Not prevalent; Supposed copycat variant
10-May-04	WORM_CYCLE.A	LSASS worm (similar method as SASSER) that terminates SASSER and NETSKY processes; Creates the same mutex as SASSER variants; Political statements in worm body implies mature author
13-May-04	WORM_DABBER.A	SASSER parasite - propagates into systems infected with SASSER using existing FTP server run by SASSER; Deletes SASSER registry entries
17-May-04	WORM_BOBAX.A	Plain LSASS worm; Similar routine to SASSER
17-May-04	WORM_KIBUV.A	Plain political LSASS worm; Similar routine to SASSER
17-May-04	WORM_KIBUV.B	Multi-vulnerability malware
24-May-04	WORM_KORGO.A	1st KORGO variant; LSASS worm with similar routines
4-Jun-04	WORM_KORGO.D	KORGO variant that first does anti-SASSER registry deletions