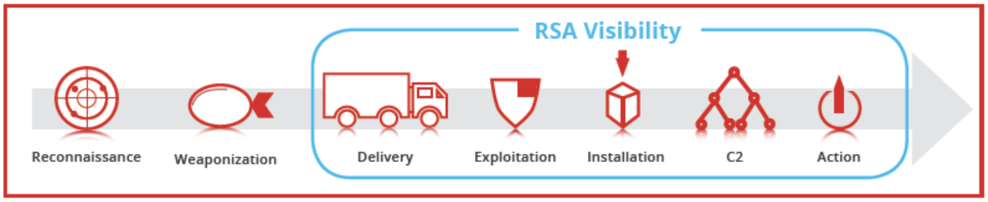
RSA

USE CASE

Malicious Protocols: Gh0st RAT

See everything, fear nothing threat solution series



What is Gh0st RAT?

Gh0st RAT is a popular example of a Remote Access Trojan used by attackers to control infected endpoints, originally attributed to threat actor groups in China. Gh0st RAT and its variants are still some of the most widely used RAT tools in existence due to their effectiveness. Once installed, Gh0st allows an attacker to take full control of the infected endpoint, log keystrokes, provide live webcam and microphone feeds, download and upload files, and other powerful features. Another feature of Gh0St RAT is the ability to obfuscate the client-server communication using a proprietary network protocol. This is wrapped up with a number of intuitive graphical user interfaces to make malicious remote control simple.

A typical attack scenario

The scenario for attacks using Gh0st RAT (or any RAT, really) follows a very typical targeted malware lifecycle. One example of how this might work is as follows:

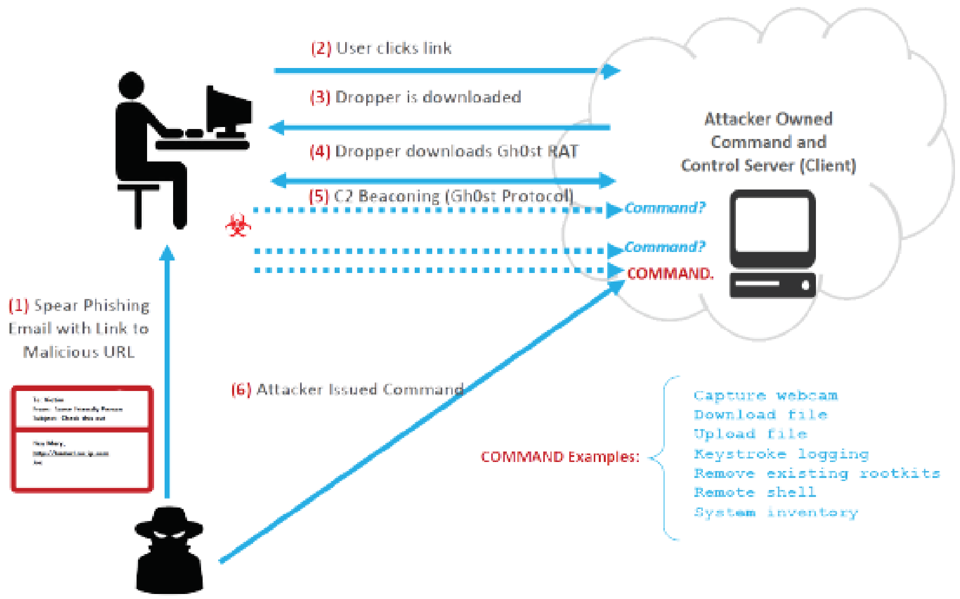
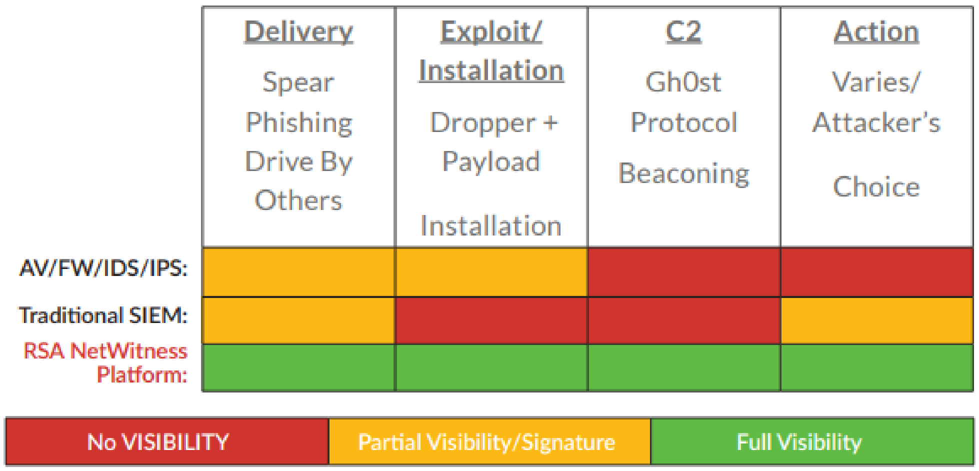


Figure 1 – Example Gh0st RAT attack scenario

Detection and response

Signature-based tools focused solely on log data lack the deep visibility into both the network and endpoint required to successfully track down attacks using Gh0st RAT. A motivated attacker can obfuscate or compile unique payloads to make detection of the delivery, exploit, and install phase extremely difficult. Visibility deep in the network is required to understand and alert on network traffic exhibiting features of Gh0st RAT C2 traffic, and deep in-memory endpoint visibility is required to track down evidence of the malicious binaries. The following chart contrasts the visibility by attack stage into an attacker's tools, tactics, and procedures (TTPs) provided by traditional tools with the RSA NetWitness® Platform:



Gh0st RAT visibility with the RSA NetWitness platform – details

Key solution: RSA NetWitness packets, RSA NetWitness endpoint

Out of the box, via RSA Live, RSA NetWitness Logs and Packets contains a network parser that can understand the C2 trafficindicative of Gh0st RAT variants. This provides a very simple mechanism for alerting on potential infections and guiding the remainder of an investigation. In this example, the analyst sees the following risk indicators within RSA NetWitness Logs and Packets:

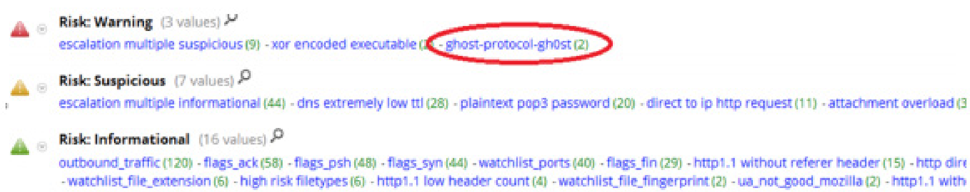


Figure 1 – RSA NetWitness logs and packets detects Gh0st protocol network traffic

Drilling further into the actual flagged events by reconstructing all relevant sessions, the analyst can see the string "Ghost" within the Hex view packet payload, which is the "magic word" for the default Ghost variant:

Figure 2 – Reconstruction of Ghost payload in RSA NetWitness logs and packet

Highly suspicious in itself, the analyst then wants to confirm the infection and glean further details into the exploitation and installation phase of the attack. The first thing noticed is the relatively high score of the suspicious machine, ACER-PC:

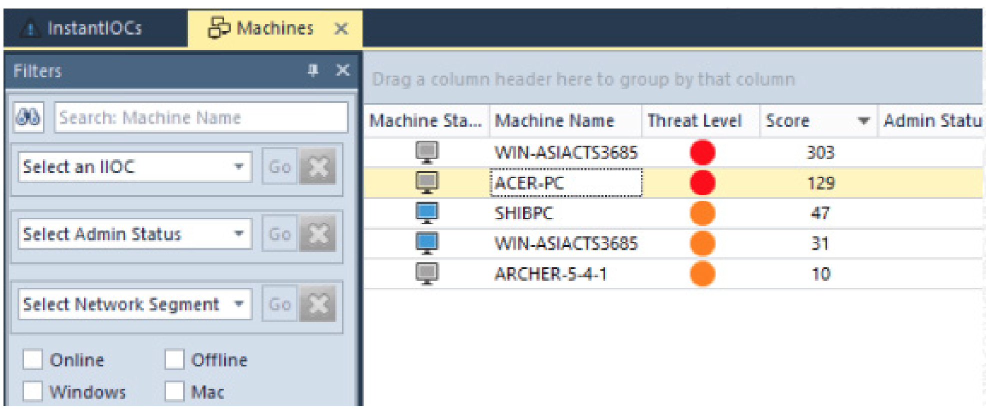


Figure 3 – RSA NetWitness endpoint showing ACER with a high (suspicious) score

Drilling deeper, the analyst notices behavior very typical of Gh0st RAT installations. RSA NetWitness Endpoint quickly identifies three malicious binaries, one of which is highly suspicious (FastUserSwitchingCompatibilityex.dll):

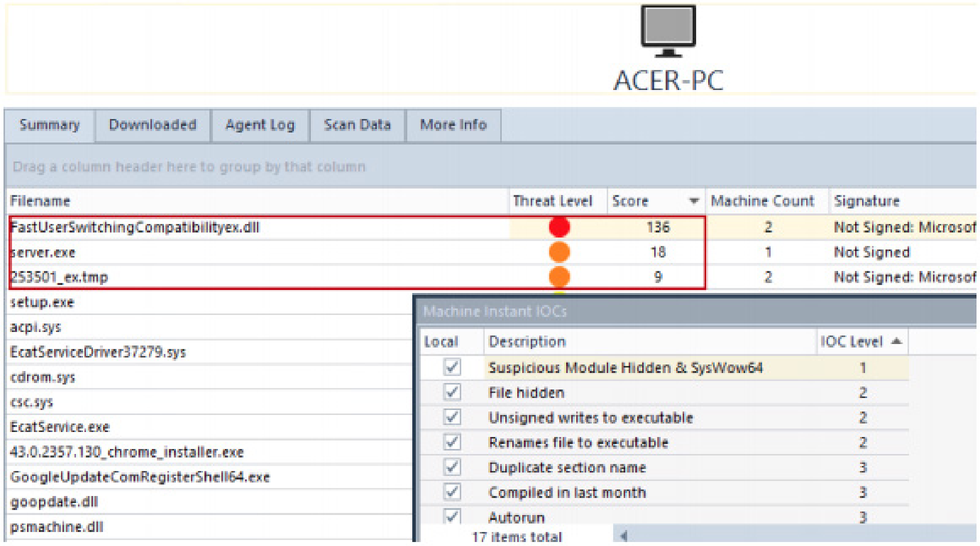


Figure 4 – Suspicious modules running on ACER-PC

RSA NetWitness Endpoint also provides the ability to search for these files on other machines in the organization's network to determine whether other hosts have been impacted by the same type of attack. Here, the analyst was able to identify another machine with the same suspicious binary which warrants further investigation into that host to understand the footprint of this particular attack across the organization for potential lateral movement that took place:

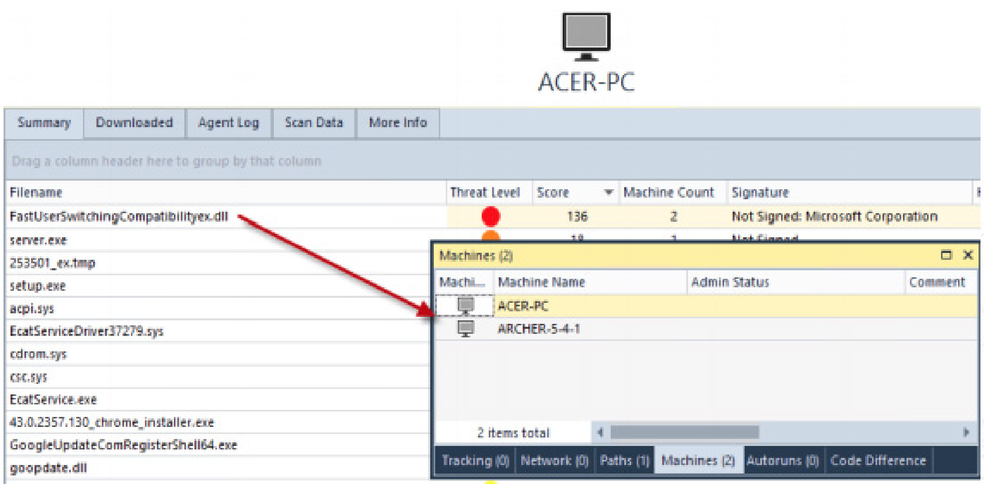


Figure 5 – Take note of other computers with the same module

In addition to the two malicious binaries running on the system, RSA NetWitness Endpoint correlates the Gh0st protocol traffic seen within RSA NetWitness Logs and Packets destined to 192.168.1.135 (the attacker IP address and C2 server) on port 8080:

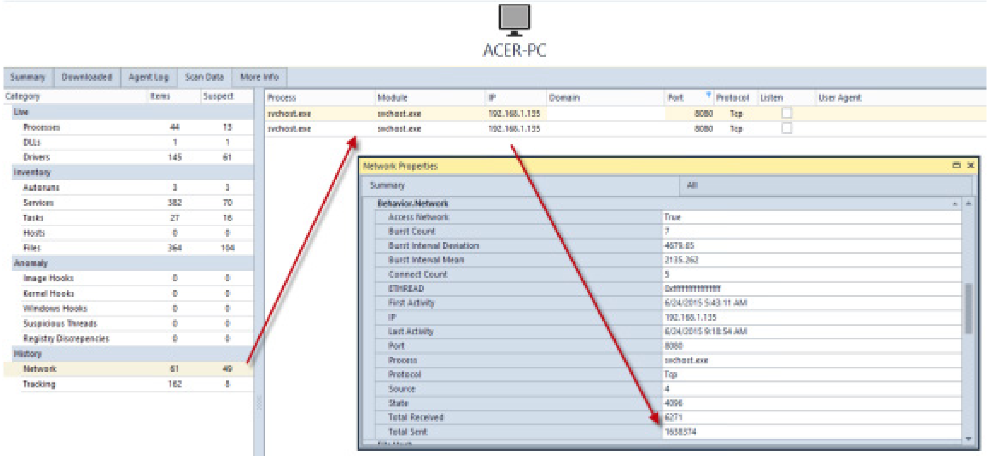


Figure 6– Identify process responsible for Gh0st C2 traffic

The analyst can also see that svchost.exe is responsible for the offending traffic, and can also glean details into the properties of the network session, including the frequency, protocol, and ratio of bytes sent to bytes received. In this instance, the analyst notices a high ratio of bytes sent vs. received, which is typical of malicious traffic and potential data leakage.

The analyst can then pivot into the remaining scan data and look for any other confirmation of infection. Very quickly they can see svchost.exe is responsible for loading FastUserSwitchingCompabilityex.dll, the most suspicious binary

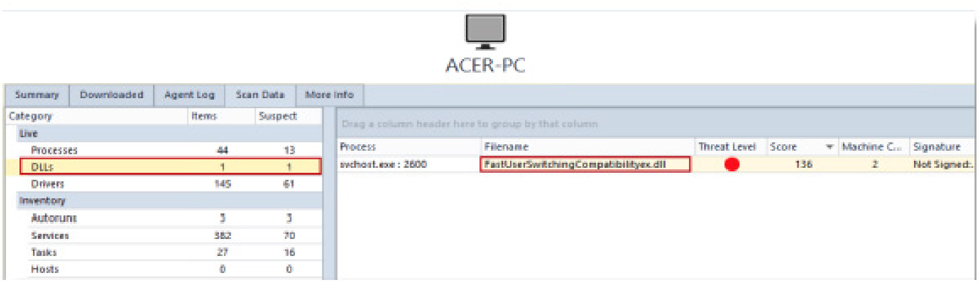


Figure 7 – Scan data showing svchost.exe loading a suspicious module

Now that the infection is confirmed, the analyst must perform the rest of the investigation. This involves further analysis of network traffic, looking for lateral movement (investigating the second machine with the same DLL), and possibly trying to attain attribution for this attack. One of the first things the analyst does is rewind the tape for the infected host to see if the delivery mechanism can be pinpointed and used to move detection up the attack chain in future attacks. This is done within RSA NetWitness Logs and Packets by reconstructing network traffic to and from our victim machine for the time prior to the infection. Doing so reveals a few interesting details, including this email message that has been automatically reconstructed:

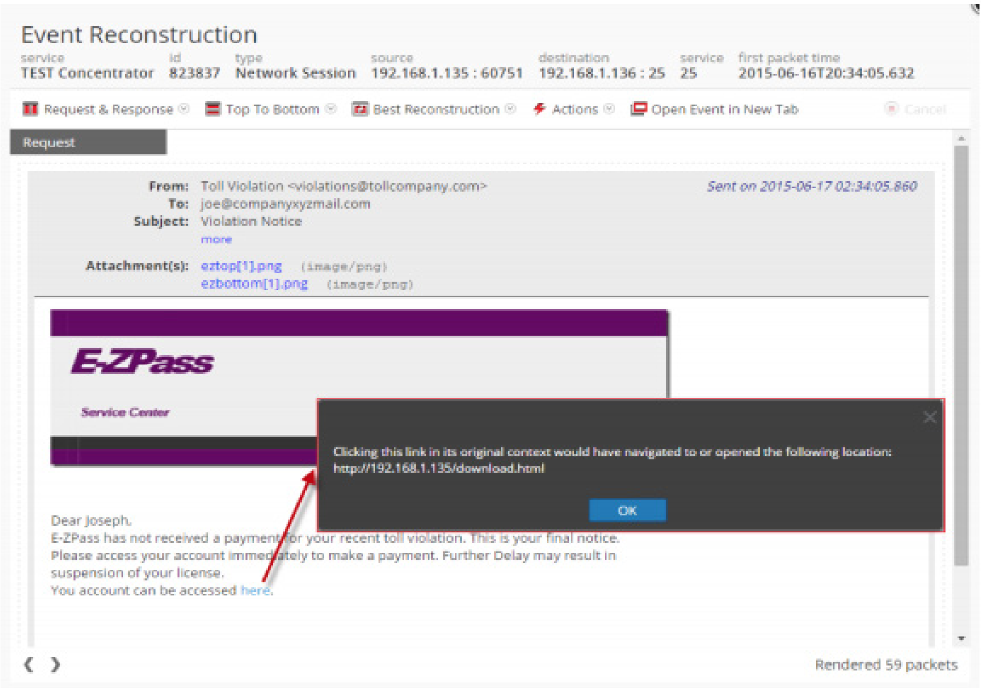


Figure 8 – Suspicious email received on the victim machine

To see whether the user clicked, the analyst can query for any HTTP sessions to http://192.168.1.135/. Doing so produces evidence of suspicious files being downloaded, which may indicate an initial infection vector:

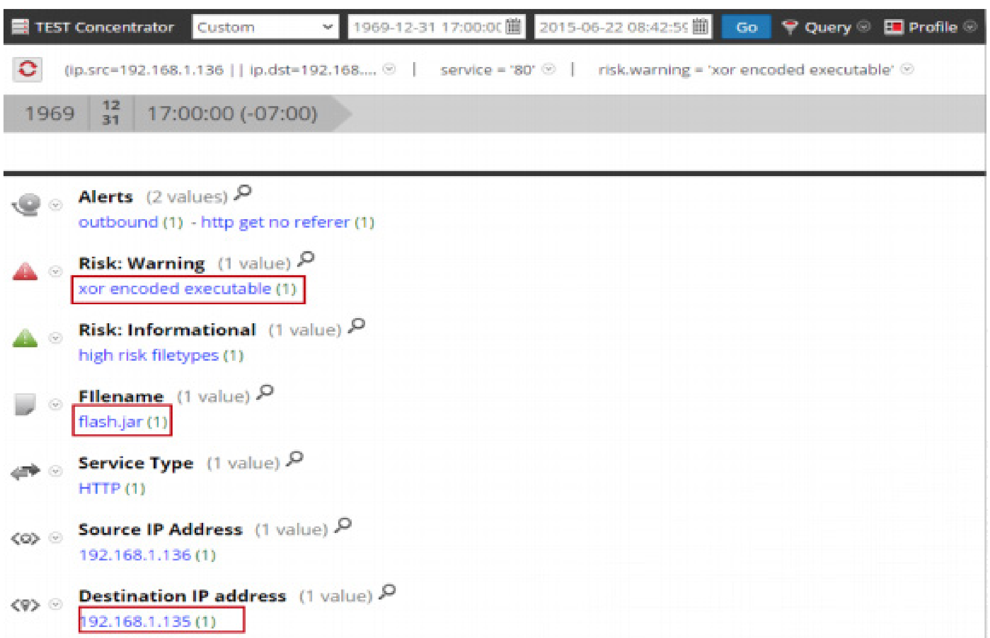


Figure 9 – Evidence of suspicious file downloaded after link clicked

Now that the analyst has reasonable evidence that the endpoint was infected with Gh0st RAT, and evidence as to how it was infected, they can attempt to understand more about the impact to the victim machine(s). Reading up on Gh0st, the analyst determines that the traffic on port 8080 represents the control commands being sent to the endpoint from the control server. This is accomplished by extracting the communications in a common pcap format and decoding it with the information learned in the investigation (in this case, the magic word "Gh0st"). Here, the analyst is able to see the adversary launching a remote shell session and what commands were run:

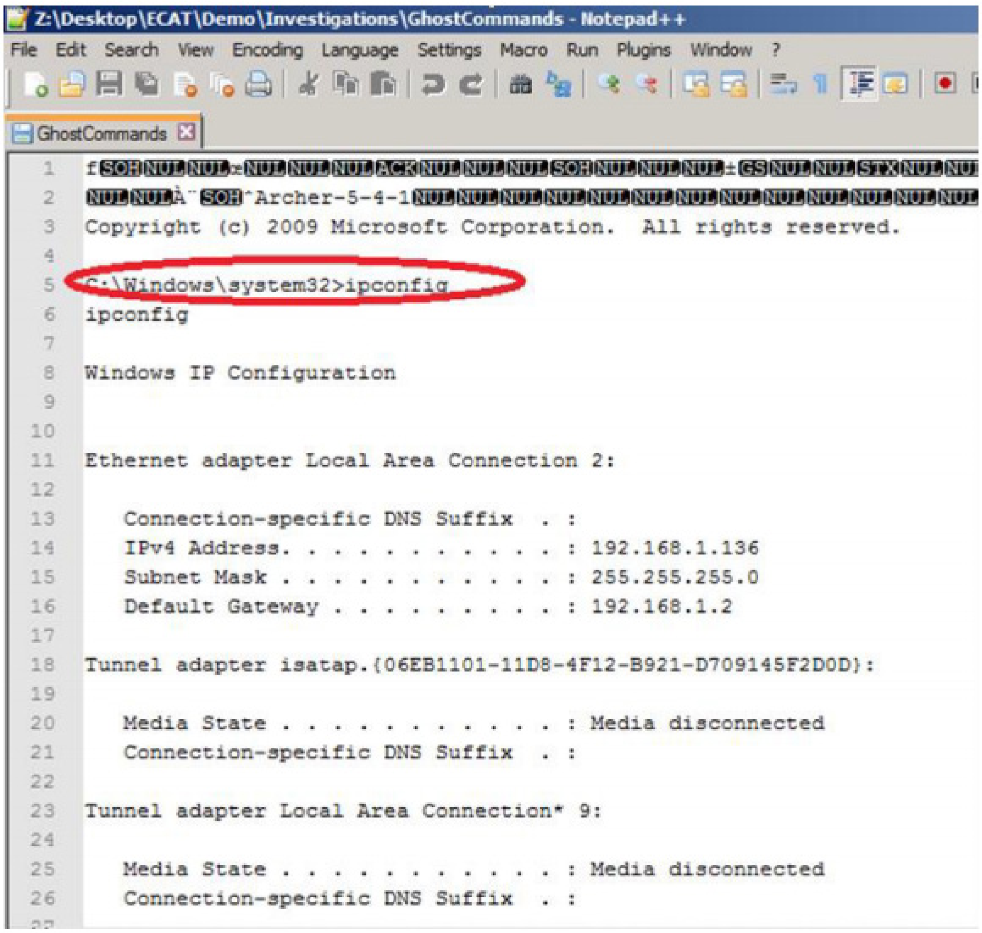


Figure 10 – Understanding what the attacker did on the victim machine

References

McAfee Gh0st analysis: http://www.mcafee.com/ca/resources/white-papers/foundstone/wp-know-your-digital-enemy.pdf

VOHO watering hole using Gh0st: [http://blogs.rsa.com/will-gragido/lions-at-the watering-hole-the-voho-affair/](http://blogs.rsa.com/will-gragido/lions-at-the-watering-hole-the-voho-affair/)

Cyber Kill Chain: http://www.lockheedmartin.ca/us/what-we-do/information-technology/cyber-security/cyber-kill-chain.html

About RSA

RSA, a leader in cybersecurity and risk management solutions, provides organizations with technology to address challenges across security, risk management and fraud prevention in the digital era. RSA solutions are designed to effectively detect and respond to advanced attacks; manage user access control; and reduce operational risk, fraud and cybercrime. RSA protects millions of users around the world and helps more than 90 percent of the Fortune 500 companies thrive and continuously adapt to transformational change. For more information, go to rsa.com.