;login:
DECEMBER 2013 VOL. 38, NO. 6
ELECTRONIC SUPPLEMENT

Conference Reports

LEET '13: 6th USENIX Workshop on Large-Scale Exploits and Emergent Threats

Washington, D.C.
August 12, 2013

Summarized by Jason Britt and Rik Farrow

Opening

Summarized by Rik Farrow

Vern Paxson opened the workshop by telling us that there were 24 submissions and ten were accepted. LEET had a small program committee (six), and decided to include three invited talks, which turned out to be some of the attendees' favorites.

Vern explained that each paper presenter had just ten minutes for the presentation, and the next 15 minutes were set aside for question and answer.

Economics/Business

Summarized by Jason Britt (jrbritt@uab.edu)

Attacker Economics for Internet-Scale Vulnerability Risk Assessment

Luca Allodi, University of Trento

Luca began his presentation by outlining his goal of changing vulnerability risk assessment methodology to something that is more Internet scale to allow better quantitative assessment of vulnerability. However, the methodology does not work for targeted attacks against specific organizations, but does work for widespread untargeted attacks. CVSS doesn't do a great job of identifying high use attacks seen in the wild. Instead the likelihood of an exploit's use should be based on the expected utility of the exploit. Luca then presented several alterations to the CVSS model.

Someone asked, when an attack tool is low cost what does it mean? Luca responded that it represents the likelihood that use will increase as cost decreases, that is, the exploit gets older and is used more often. Stefan Savage (UCSD) asked if Blackhole uses a new business model. Blackhole reinvests in attacks to create zero days for broad use. Luca responded that it does, but low-cost tools will still be an avenue of attack and still need to be addressed. Chris Grier (UCB) asked whether there is a correlation between corporate payment for exploits and attack-tool cost—for example, Google. Luca said that his feeling is the price of black market exploits do not correlate with corporate payments for exploits because corporate payments are for attacks specific to the corporation and not broadly useful in the general black market. Vern Paxson asked whether the makeup of the defender affects the framework. Luca responded that defender makeup does affect the framework. This model is attacker oriented, but does need to take into account defender makeup. Vern then asked how the model parameterizes defender makeup. Luca replied that the process to generate the model needs to change instead of weightings. Vern continued to press Luca about defining the temporal and environmental parameters. Luca responded that this is incorporated into some of the base numbers and weights in the base core model. Even in other models this is a problem as it requires a large investment to tailor for specific defenders. The final questioner wondered whether the data perspective skewed the results. Luca responded that they try to pick populations out of each data set to determine something about the population instead of the individual targets.

There Are No Free iPads: An Analysis of Survey Scams as a Business

Jason W. Clark and Damon McCoy, George Mason University

Jason began by saying a survey scam is the "win something if you take this survey" ad you may see on Facebook and other social media. Jason said that cyber criminals are focusing on Facebook, trying to lead users to off-site places to collect information from users for marketing purposes.

Jason said that 1,700 survey scam/spam URLs were gathered from several Facebook spam feeds and manually sorted into one of several categories. The individual spammers were identified by the pub ID of the ad, and the gate ID is the offer to be displayed. Fifty percent of the URLs were tracked to one of four ad networks. To join some networks you just need an email and other credentials, others require online interviews, and some require geographic locality. Jason said the pub ID was also used to track spammer growth over time as the pub IDs appear to be sequential.

Jason also said that the revenue each spammer generated was estimated to be relatively low. Users were lured into clicking on ads using interesting wow factor content or sexual content. Several conclusions were reached as a result: 73% of working spam URLs in Facebook spam were monetized survey scams, and ad networks aren't removing abusive marketers.

Vern Paxson asked what happens to the victim. Jason said a victim's personal information is spread wider for marketing, including the use of their Facebook wall. Chris Grier asked, do they tell you as a spammer what you are advertising? Jason said that you can pick offers to advertise; they believe the affiliates know on some level what is happening. Someone wondered why there were so many dead links. Jason said this was the result of the fetching mechanism and takedowns of ad sites. Vern Paxson asked how they know the pub ID is always going to increment by one. Jason said this is not a guarantee, but seems very likely. Vern asked about the conversion rate of people clicking on the ads. Jason said the conversion rate is about 2%.

Invited Talk: Bitcoin in Cybercrime

Stefan Savage, University of California, San Diego

Stefan began the talk with a description of the Bitcoin system. Bitcoin is a purely online currency and payment system that uses a decentralized P2P network, where each Bitcoin user keeps a record of all Bitcoin transactions. Bitcoin wallets are public keys, not tied to any personally identifiable information, and Bitcoin transactions are designed to be irreversible. Bitcoins can actually be used to purchase things. Vendors accepting Bitcoin are increasing, partially due to services such as Bitpay. Bitcoin has risen in value, but is very volatile.

A Bitcoin is a chain of transactions that record where it's been from creation till now. The chain consists of input bitcoins and output bitcoins and accounts. This is the provenance chain of Bitcoin. To make a transaction you use your Bitcoin hash signed with the recipient's public key.

New bitcoins are mined. Mining is done by finding SHA256 collisions that begin with a certain number of zero bits and the reward is a set number of bitcoins, declining asymptotically as the maximum number of 21 million bitcoins is approached. To use bitcoins, you don't need to know any of this. Bitcoins can be used for criminal activities—avoiding payment cards, direct wealth creation through theft—and have no overarching central authority to shut down or track.

Stefan then proceeded to talk about the realities of Bitcoin. Mining bitcoins is not cost effective because of electricity, but botnets have free electricity. CPU has not been factored into the cost of renting bots, and this is where bitcoin mining on botnets comes in.

Inter-criminal payments are anonymous and irreversible for malware purchase and affiliate payments. However, central payment exchanges that convert bitcoins to other currencies can be shut down. There are a few exchanges to convert to and from bitcoins. These are central authorities that can be forced to comply. Bitcoins can be tracked back to exchanges and conversion to real money to track down criminals.

Someone asked, why not create one wallet per transaction? Stefan responded that you can, but it still doesn't help with anonymity. Because you have to have "change wallets" and reaggregate your coins into a single wallet. Rik Farrow asked about pseudonymity, and Stefan answered that there is research he is involved in that shows how wallets can be aggregated and associated with exchanges and other online entities. Exchanges are particularly vulnerable to being forced by governments to provide information about people who cashed in bitcoins. Someone else asked if double-spending is possible because of the distributed process, and Stefan said that it could happen, as there are delays, sometimes in the tens of minutes, between a transaction and that transaction being committed to a chain.

Botnet Analysis and Evolution

Summarized by Jason Britt (jrbritt@uab.edu)

FuncTracker: Discovering Shared Code to Aid Malware Forensics

Charles LeDoux, Arun Lakhotia, Craig Miles, and Vivek Notani, University of Louisiana at Lafayette; Avi Pfeffer, Charles River Analytics

Charles LeDoux started his presentation by describing how malware research has changed from only determining whether software is malicious or benign to adding more complex questions such as who wrote the malware, why, and why is the malware targeting a particular entity. To try and answer these new questions, individual pieces of malware need to be related and treated as a single piece of a puzzle.

Charles talked about how relating malware can be accomplished through relating based upon shared code. This has been done before in analysis and realization of shared code bases in Stuxnet and Duque. Existing approaches cluster related malware on whole binary comparisons, which are not scalable. The goal is to create a scalable method to extract sets of non-insignificant code pieces to represent the malware for clustering. A functional granularity is used, each functional representation is hashed, and the hash is used to represent the function. The functional representation is generated using the GenCode style method of abstracting registers and constants, then mapping the effect the code block will have on the abstract registers.

The functional representations are stored and related, and the relations are searchable through three different searches. The three searches are traversal, shared attributes, or both. The searchable relations can be used to find binaries that have a particular behavior. Further improvements include a need for a more comprehensive evaluation and extending hashing to make it less fragile.

Someone asked whether packing was considered. Charles responded that it was considered, but it was out of scope for this project. You would have to unpack before being able to apply FuncTracker. Someone asked, why not link to an intermediate language before evaluation? Charles responded that GenCode essentially performs this, but further work could be done. Someone asked, what does and doesn't GenSemantics handle and are you happy with GenSemantics? Charles said GenSemantics does need improvement mostly because it doesn't handle weird x86 commands.

Botnet Triple-Channel Model: Towards Resilient and Efficient Bidirectional Communication Botnets

Cui Xiang and Zhai Lidong, Chinese Academy of Sciences; Zhang Yuxiang, Xi'an Research Institute of Hi-Tech Hongqing Town; Guo Yunchuan and Liu Chaoge, Chinese Academy of Sciences

Cui opened his presentation by giving a history of the first three generations of botnets. The first generation of botnets had a static centralized command and control topology. First

generation botnets suffered from a single point of failure. Second generation botnets had a decentralized command and control topology with a peer-to-peer–based network. Second generation botnets are vulnerable to index poisoning by redirecting a bot to a peer you control. The third generation of botnets utilized a dynamic centralized topology.

Cui then described a framework for a third-generation botnet that is resilient and scalable. The framework utilizes a triple channel model for communication. The three channels are command, registration, and data channels. Most third-generation botnets utilize resilient one-directional communication channels. The command channel uses a two-phase process. The first phase publishes commands to the dynamic command and control elements. The second phase involves bots attempting to locate command elements and receiving the issued commands. The registration channel uses domain flux to hide and to make the botnet sinkhole resistant. The registration channel allows bots to download registration information that allows them to communicate over the other two channels.

The data channel also uses a two-phase approach. In the first phase, bots upload files to randomized URLs in the cloud. In the second phase the bots' randomized URLs are communicated to the command and control elements over the data channel.

The session chair asked what about their motivation for this research. Cui responded that they wanted to create a new C&C botnet framework. Someone asked about their threat model for botnet takedown, and were they concerned about the privacy of the file they uploaded to the cloud? Cui said that the data channel idea and storage topology on the cloud is new and as such he is unsure. The session chair asked how they would take down a botnet that utilizes the triple channel architecture. Cui responded that using network definitions might help, but that it is a hard problem mostly because the botnet acts like three separate networks.

SinkMiner: Mining Botnet Sinkholes for Fun and Profit

Babak Rahbarinia, University of Georgia; Roberto Perdisci, University of Georgia and Georgia Institute of Technology; Manos Antonakakis, Damballa, Inc.; David Dagon, Georgia Institute of Technology

Babak started by describing what a sinkhole is and does and the motivation for the sinkhole investigation. Sinkholes are used to take over a botnet and or measure it through victim enumeration. Cataloguing sinkholes is useful as it allows for the measurement of command and control (C&C) domains, and helps to avoid the collision of multiple benign takeover attempts, such as by law enforcement.

Babak went on to describe how sinkholes were catalogued. A large passive DNS database was used to travel back in time to get a list of all domains resolving to particular IPs used as sinkholes. This showed where the domains went after pointing to a particular sinkhole IP. Preliminary labeling of domains was performed manually by looking for known sinkhole IPs. Then the passive DNS database was used to determine the DNS server used to resolve the domain, sometimes showing a DNS name that was saying it was a sinkhole. Other IPs were identified by finding lots of invalid domains pointing at them in which the IPs were limited to organizations known to perform sinkhole operations. This provided the base of sinkhole IPs. Next, the known sinkholed domains were examined to see the other IPs to which they resolved. This new IP was labeled as a likely sinkhole IP and the analysis was repeated to find more likely sinkhole IPs.

Babak then described the results. Of the 22 known sinkhole IPs, 39% of the domains pointed at them shifted to new IPs. The recursive analysis technique allowed for the identification of another 87 likely sinkhole IPs.

Someone asked where their passive DNS data came from. Babak said the passive DNS data came from Comcast, based upon the entire Comcast network, through their relationship with Damballa. The session chair asked why the sinkhole IPs moved around. Babak said the sinkhole operators reregister domains to prevent their release and that one IP can be used to sinkhole many botnets. Vern Paxson asked, when IPs cross ASNs could they know whether the IP was still being sinkholed? Babak responded that in many cases the new IP is another organization known to sinkhole. Because it is in the wild and people don't always get back to you, the problem is developing a ground truth of known sinkhole IPs. Since people don't want their IP to be a known sinkhole, they may not respond.

The real problem is that information sharing is not big in security currently. The last question was, are the sinkhole operators trusted addresses? Babak said that opening a sinkhole is a delicate matter, because of network reputation degradation. Hence, they concentrate many domains on a few IPs to help prevent the address reputation damage. Vern Paxson commented that Microsoft does seizures by the book.

Invited Talk Testing, Testing, 1 2 3: The History and Challenges of Testing Anti-Malware Software

Mark Kennedy, Anti-Malware Testing Standards Organization and Symantec Corporation

Mark started his talk by going over some definitions and background. The Anti-Malware Testing Standards Organization is made up of testers, academics, and vendors and was started in 2008. The types of testing have changed over the years. Several types are used today, including dynamic testing and real world testing. Dynamic testing came about because a whole suite of tools did not work statically. When performing testing the test needs to match the real-world use of the tool being tested.

Mark then addressed the challenge of sample selection. There are several challenges in choosing samples for testing, such as

one versus many families, inclusion of grayware, and inclusion of junk samples that do not run properly on the configuration being tested. Selecting samples from families is difficult. While there are only a thousand or so active families on any given day there are millions of examples. Also, targeted attacks are difficult to represent as they may flood certain defenders but not others. Grayware inclusion can cause problems in sample selection. Grayware can be hacking tools used by malware that can also be used legitimately, adware, or other unwanted but not illegal software.

Legal challenges can cause problems on the grayware front as software producers many sue large security vendors for labeling their software as malicious. Transparency is needed to resolve these issues. The setup and items tested need to be known as well as the testing results.

Mark next listed the nine principles of good testing and covered issues associated with some of them. To prevent real-world damage when testing, slow Internet connection speeds can be used to limit the damage that can be caused. Performance testing should be fair and balanced. For example, a machine shouldn't be over-infected resulting in a non-real-world scenario. Testing should be performed over extended periods of time. Information used during testing such as vendor ground truth should be collected immediately, when the sample is received, but should also be re-gathered at a later date as vendors are often on the bleeding edge and, given time, will catch up. Also, you need to intimately question your testing data set. Detecting bad things is relatively easy, but in doing so you don't want to shut down good things.

Mark closed the presentation noting full documentation covering testing best practices can be found at the AMTSO Web site and that there is an upcoming conference in Montreal on malware testing research.

The Untrustworthy Web

Summarized by Rik Farrow

The Devil Is Phishing: Rethinking Web Single Sign-On Systems Security

Chuan Yue, University of Colorado, Colorado Springs

Chuan looked at the actual use of single sign-on (SSO) systems from the perspective of their implementation. He focused on OpenAuth 2.0, but there are other systems by Microsoft, Facebook, and so on. Chuan used a diagram to explain the interchange between the relying party (RP), the user, and the Identity Provider (IdP). He did a formal security analysis of some Web SSO protocols and found many vulnerabilities: for example, RP not using HTTPS, logic flaws, and vulnerabilities in many deployed SSO systems. There are no technical guidelines provided to RPs.

Chuan pointed out that the value of IdP accounts is highly concentrated: for example, a Google account provides access to many Google assets. He also said that the attack surface is very large (all Web sites using IdPs) and it is difficult for users or algorithms to detect phishing. Chuan explained that while the typical phishing attack could appear suspicious, with Web SSO, the user is already familiar with clicking a button that represents an IdP, and presenting a phishing frame that looks exactly like a popup IdP login window is trivial. Even spoofing the Extended Validation SSL icon and the correct URL are simple, as they are included in the image. He used HTML, CSS, and JavaScript to produce duplicates of the actual popup windows.

Chuan performed a small user study (28 people), 22 of whom answered that they had Web SSO experience. He had them log into their fake shopping site using Google or Facebook account information (set up for this experiment) or into sears.com using an IdP. A handful of participants said that they thought the login page was spoofed, but none actually attempted to click on the EV-SLL icon and the HTTPS URL address, suggesting that people rely more on look-and-feel to identify the credibility of Web sites.

Rik Farrow asked what he should tell old people to look for. Chuan suggested that they should try and move the popup window outside of the browser window. Someone else asked whether he saw the phishers primarily going after identity theft or financial theft. Chuan said the focus appeared to be more on identity theft than on the financial sites, where the possibility of phishing detection is higher. The session chair asked if the test users that were not deceived actually typed in their credentials, and Chuan said that those eight didn't think it was real because they noticed minor discrepancies in the popup window.

Image Matching for Branding Phishing Kit Images

Chengcui Zhang, Rajan Kumar Kharel, Song Gao, and Jason Britt, University of Alabama at Birmingham

Jason Britt explained that the goal of their work was to classify phishing pages automatically for branding phishing kits. He pointed out that phishing pages look like the real thing; what they looked for were characteristics of the phishing kit. He showed examples of JS, including email addresses found for use in Drop mails (sending off stolen credentials, as he was using a PayPal example). He then said they manually went through the page looking for identifiers, but the goal was to create an automated method to do this, looking specifically at the images.

They used color histograms (CH), created using the highest two bits of RGB to create 64 buckets, and then collected this over the entire image. They tried global CH, then local (nine zone) LCH with dimensional constraints and no background, and LCH++ with a minimum bounding box which attempts to isolate part of the image from background.

They used a collection of 56,926 kits, containing 10,130 unique images with 205 brand images and 9,915 generated images. They

manually viewed images to create ground truth, then separated the brand images into two sets, one for testing and one for training. They determined the optimum distance threshold for each of the four types of color histograms, and ran the test for true positive, true negative, false positive, and false negative. LCH and LCH++ worked the best with 96.72 and 99.08% accuracy, respectively. LCH with a grid would produce a false negative just by moving a bit of gold pixels in "Verified by Visa" as an example. So LCH was much better than Global CH; this worked well enough for kit branding strategies using image brands.

Someone asked if they noticed any difference in the types of graphic images or file formats. Jason replied that they didn't look for that, they focused on the image. Michael Bailey said that he thought he was going to do something different. Phishing has to look like the genuine Web site, so could you use this to automatically distinguish phishing sites from real sites? Jason replied that there has been a lot of work on that already and that they were taking a different tack by going after kit types. Someone else asked what inspired looking for differences in color distribution. Jason said that it's easy and fast, just looking at 64-byte vectors. Performance vs accuracy.

A View to a Kill: WebView Exploitation

Matthias Neugschwandtner, Martina Lindorfer, and Christian Platzer, Vienna University of Technology

Mattias Neugschwandtner described a technique that makes creating apps for smartphones much simpler, while at the same time, opening an avenue for attacks. All major smartphone vendors support WebView, a method for writing an app that downloads, displays, and can interact with HTML downloaded from Web servers. WebView works with third-party frameworks like Apache Cordova, and updates just require a change of Web content.

WebView enhances functionality by providing access to hardware buttons, persistent storage, camera, contacts, SMS, location, etc. But this also means no containment (sandboxing) of web content.

Mattias explained two types of attacks: placing a malicious script on the server for the app, and injecting code into the downloaded Web page, a MITM attack. Attacking a server affects all users of the app, while injecting code only affects those vulnerable to the MITM—along the network path of apps that don't require HTTPS and valid certificates. Mattias provided a couple of examples, such as "Take Weather" where you can take photos of current weather and share it. But it uses plain HTTP, has access to contact info, and has access to all Java code because Android provides a bridge between JS and Java.

Mattias provided a second example, Chinese Foursquare, which has permission to access external storage and can install packages. The app authors have overwritten the SLL error handler with one that accepts any certificate! Using Andrubis, they found that 30% of apps were enabling the Java bridge; 27% of these apps were using unencrypted HTML or JS or lax SSL handling. They also found permissions for SMS in 11%, installation (read/write) in 60%, and privacy (contacts, location) in 76%.

The suggested mitigation is to use HTTPS and correct certificate handling. That is, signed certs and cert pinning, but WebView is targeted at inexperienced developers. Android 4.2 (Jelly Bean) introduced the @Javascriptinterface annotation, but this only prevents reflection attacks.

Jason Britt asked whether this implies the ability to interfere with other applications that may be running, like a banking app on his phone. Mattias said that if you have install permissions, you can run whatever app you want. Stefan Savage said that we've run into this again and again—half-a-dozen ways of breaking functionality X. He wondered whether there wasn't a better solution we can use than publishing papers about this. Mattias responded that this is a question for the whole room and a question that has no easy answer, once an API has been offered. The session chair said that if you have an auto-update capability, a network adversary can install an app that can be reinstalled. Stefan wondered why we provide a set of capabilities that a developer can get into trouble with. Brian Warner (Mozilla) said that because you can update the HTML, either by replacing it on the server or inserting it, it's just as big a problem. Mattias said that in this case, it is not just the server, but all the people who have installed the app who will be affected (500k in the Chinese example, because of auto-update).

Modern Denial-of-Service/Threats

Summarized by Rik Farrow (rik@usenix.org)

Recent Advances in DDoS Malware

Jason Jones, Arbor Networks

Jason began his invited talk by telling us that he got involved with this project at Arbor because he liked math and graph theory and wanted to use it for analyzing malware. Arbor has built a massive malware analysis system they call ASERT Malware Corral, and are pulling in 100,000 samples per day, with a focus on malware that can do DDoS. Last year they found 567 families of malware.

DDoS has continued to be a threat, with the attacks on SpamHaus, financial and news servers, and some politically motivated attacks. Some things haven't changed since DDoS first appeared around 2000: basic GET/POST, SYN/connection floods, UDP floods, and the use of IRC for botnet command and control (CnC). They also see lots of HTTP CnC, with obfuscated commands, but little HTTPS/SSL so far.

Jason provided some examples of current bots that can perform customized HTTP attacks, like Athena, which can include a list of legitimate user agent strings, Armageddon2, which uses a list

of provided HTTP headers, and DirtJumper, with its ability to fill in POST requests with parameters, like random thousand character-long names for forms. Athena performs some obfuscation, like using a random number generator to select headers.

The attack on SpamHaus used DNS amplification, which relies on open DNS resolvers, but this has been rare so far. Jason expects to see this incorporated as a feature soon, as it is too successful and easy not to be included.

The session chair, Mike Bailey (University of Michigan), asked how they know they are getting good samples. Jason replied that they trade with some AV companies. Someone else asked how they identify malware. Jason said they find more than they can check, so they groom their samples, using tools like Yara (http://code.google.com/p/yara-project/) to filter out stuff they've already seen. The real problem is that bots have the ability to be updated. Rik Farrow asked about shutting down open resolvers, and someone suggested he look at the OpenResolver project.

Understanding the Emerging Threat of DDoS-as-a-Service

Mohammad Karami and Damon McCoy, George Mason University

Damon McCoy began by describing booter services, cheap services you can find with Web searches that perform DDoS. For example, you can provide someone's Skype, Steam, or Cloudfare names and the booter service will convert the name into an address and launch a DDoS attack. A typical attack lasts an hour and costs $22 or less if you have a monthly subscription to the service.

Leaked data from the TWBooter2 service showed that this service had 312 users, 11,174 victims, and launched 48,844 attacks over the service's several-month life. TWBooter2 used from three to nine servers hosted in The Netherlands to run its Web scripts and control proxies and relays, and could launch SYN floods at 40 Mb/sec. Their HTTP flooding used 26,296 proxy servers. Most of the attacks came from just six users, with game servers, game forums, and other booter services the most popular targets. Another booter service, Asylumbooter, had 5,622 subscribers and launched over 500,000 attacks during a 16-month period.

Someone asked if these services advertise the amount of bandwidth, and Damon answered that not all do, or that the numbers can be verified, but they appear to range from three to 65 Gb/sec. Mike Bailey asked how much booter services customize their attack code, and Damon replied that they mostly just use source code that has been leaked, so are very similar. Someone else asked how people pay, and Damon listed PayPal, Bitcoin, and WebMoney to a lesser extent. The same person asked if there was customer support, and Damon said that there was, they had tried it. Mike Bailey asked why these servers remained up so long, and Damon said that they had been hidden behind proxies.

Invited Talk: Stepping P3wn3: Adventures in Full-Spectrum Embedded Exploitation and Defense

Ang Cui, Red Balloon Security and Columbia University

Ang Cui reprised a talk he had given at Black Hat this year about exploiting embedded systems, like HP Laserjets, Cisco routers, and VoIP phones. Both an entrepreneur and a fifth-year graduate student, Cui explained how he scanned the IPv4 address space and found 90,000 vulnerable printers. When he rescanned these printers 14 months later, only 7.42% had been patched, saying that people generally don't patch printers.

Cui described an attack designed to get inside a company's network. The attack begins with a resume containing multiple exploits for printers. If someone prints out the resume on a vulnerable printer, the printer fetches and installs a more advanced program that is used as a proxy for more attacks. The next step is to scan the internal network, then attack other embedded devices. They discovered that a $150 HP Laserjet printer could send out 15,000 packets/second, and could be used to DoS low-end routers like a Cisco 2821. Most VoIP phones use Linux, and old versions too, so they can be attacked. Post-attack strategies include reflashing embedded devices so they will include persistent backdoors. They can also enable VoIP phone microphones so they can listen in on conversations even while the phone is on-hook.

Cui said that they do have a defensive strategy, which involves installing hooks into the firmware. These hooks call into software they also install, which he called Symbiotes, that can monitor systems against code injection attacks. Mike Bailey asked what they do when they detect an attack, and Cui said they can notify the owners to reflash and reboot the attacked devices. Mike then asked if they can detect dynamic code (rather than in text regions), and Cui said they can. Cui ended by providing a link to his slides, http://redballoonsecurity.com/slides.pdf, but I wondered about downloading them.