Wireless malware propagation: a reality check

Stefano Zanero*
Dipartimento di Elettronica e Informazione
Politecnico di Milano, Milano, Italy
stefano.zanero@polimi.it

June 29, 2009

In recent years, a number of authors (including myself, I confess) began to work on the concept of security attacks, and in particular of the propagation of malware, through wireless communications protocols. The idea challenged and thrilled us, mainly because it brought back the concept of physical and geographic interaction in the attack scenarios.

It was fun to design covert attack devices and evaluate the exposure of the Bluetooth user's population to them [1]. It was also fun, I bet, for other authors of papers on wireless router subversion [2] and malware propagation on WiFi networks [3] to envision their attacks.

However, it is my opinion that the latest "developments" on these threats are stepping progressively away from reality and into an abstract, academic world of their own, which may be just as fun, but which should be brought back into perspective when assessing the actual risks related to these scenarios. Let's see why, with two distinct examples.

The unlikely router contagion A recent paper [4], which received some news coverage outside the scientific circles, describes the propagation of a virus over a set of overlapping wireless LAN networks in an urban area. The wosk is theoretically and mathematically interesting, but sadly it describes a type of malware which is highly unlikely to appear in the real world, for a number of reasons.

In their introductory assessment of the prevalence of wireless worms, the authors mix together very different threats. They refer to the attacks described in [5], which actually come from the the Internet side of the router connection, and need an interaction with a client on the WiFi network itself. Something similar happened in the wild with the "Zlob" trojan [6] (which, by the way, attacked wireless and wired routers indifferently, to stress that the "wireless" component

*The author gratefully acknowledges the help of Dr. Kostas Anagnostakis (Institute for Infocomm Research, Singapore), and of Polimi colleagues Prof. Carlo Piccardi, Dr. Renato Casagrande, and Guido Salvaneschi. The opinions expressed in this article are exclusively those of the author.

was irrelevant). Zlob infected clients on the LAN, and then tried to guess administration passwords using a built-in list of default username/password combinations. If successful, it would then alter the DNS records to perform a MITM attack. Similarly, the worm propagation described in [3] uses clients roaming from network to network to spread a contagion in a local metropolitan area (a so-called "wildfire" worm). All of these are very realistic attack vectors.

On the contrary, a **router-to-router** attack, such as the one envisioned in [4], is much more difficult to execute than the paper maintains. First of all, domestic wireless routers (as opposed to core Internet routers) are a very diverse family of devices. In virology (even in the biological world), homogeneity breeds danger, while heterogeneous devices are less susceptible to a digital contagion. This is evident, for instance, in the mobile phones world, where heterogeneity of devices is one of the key obstacles to virus propagation (and to benign software development, but this is a different issue). We will see more on this later on.

The authors theorize the creation of a universal bogus firmware, which is not going to happen, as should be evident to any reader with experience on embedded networking devices. It's difficult to write reliable bogus firmware for similar, but not identical, platforms: let alone for completely different ones. Of course, attackers could target a popular and easy to customize model (such as the Linksys WRT54G, for instance): but this will of course reduce the number of possible targets, way below the numbers the authors use. The only known example of firmware malware in the wild to my knowledge was the Bluepill botnet [7], which was deployed from the Internet and once again did not deal with the wireless side of the router.

One of the reasons for attackers being so shy and not touching the wireless connectivity part is, of course, stealthiness. In order to use the radio apparatus of the router for scanning and deploying malware on other networks, wireless service to clients would need to be disrupted. This would be likely noticed by the users, and lead to disinfection or disconnection of the router, which is a big no-no for a malware author.

But taking this huge problem aside for a moment, we are left with the idea of uploading bogus firmware on open routers with a specific model. This was actually already explored more sensibly in [2], and the authors there estimated that 34% of observed routers were of the appropriate models. Reading through the paper, it becomes obvious that this is an overestimate, as it comes from a potentially biased projection of a very limited subset of identifiable routers. But still, we can use it as an upper bound: less than one third of observed routers could run an hypothetical attack firmware. Furthermore, in [2] only 16.7% of such routers are shown to have default settings, which the authors translate in an overestimated 10% of routers with no password or default passwords. In [4] this is assumed to be a staggering 50%. This brings down the targets from 50% to less than 5%: quite a show-stopper for any aggressor.

And, while we are at it, in [4] the percentage of encrypted networks is extremely low (compared to many other studies), and actually it does not even match with the website cited as a source by the author (which we report for comparison in Table 1). On the other hand, using data in [8], cracking a 104-

Table 1: Data on network protection from wigle.net (updated 5 Apr 2009)

Networks with WEP	8,405,111	48.0%
Networks without WEP	6,078,663	34.7%
Networks WEP unknown	3,008,994	17.2%

bit WEP key takes approximately 1 minute of data dumping and 3 seconds of computation using a 1.7GHz Pentium M processor and 3MB of RAM. WRT54G devices have various processors on board, but the best ones are 200MHz MIPS32 processors with 16 MB of RAM: it would seem likely that the times for cracking a WEP encrypted network are actually overestimated by the authors. Bruteforcing administrative passwords may make sense in theory, but a million-password list is impossible to use in such a setting, as it would easily exhaust the available device memory. Also, the fact that "wireless routers do not have bruteforcing protections" is not stated anywhere else, and should be demonstrated.

Dulcis in fundo, the authors assume that each infected router, in a given infection cycle, will attack only "new" routers not previously attacked. This is impossible to ensure (as it would require a level of distributed coordination of the malware which is difficult to envision without a botnet-like command-and-control structure: and if you are going to make this a botnet, you may as well infect those devices from the Internet side). This assumption, which is seemingly of small importance as it is confined in an appendix of the work, actually significantly biases the propagation data, as it removes a reinfection term that in proximity-based infections (such as this one) is overwhelming.

The net result is that while the mathematics in [4] is fun to read through, the type of malware described is highly unlikely to appear, to put it mildly. It should exploit weaknesses on a heterogeneous population, without relying on client infection; it would work only on a small fraction of routers, which are unlikely to form a connected cloud; it would disrupt the regular use of the networks, and thus be noticed. Also, the simulation parameters are far from real-world data. But the most critical question left unanswered here is why. Why should an attacker run such a complex attack, if (as shown in [7]) it is possible to obtain significantly high penetration through the wired Internet interface of routers, in an easier and stealthier way. Even then, a compromised router is of little use as a bot (since its data storage capabilities are extremely limited). Interception of user data is an interesting and worrying perspective, but then again, what is the specific advantage of creating a wireless-propagating malware as opposed to an Internet-based worm or drive-by attack to do that?

This is what happens when mathematics comes first, and actual risk assessment comes later, if ever. When I shared these remarks with the editorial board of the journal where this article appeared, I obtained an answer which actually provoked the line of thought in this article: "an argument about whether a given model does or doesn't "reflect the real world" (especially in a rapidly evolving

field as wireless) is not a good use of [our] letters pages. I don't think there is much danger that the model published will be taken as literally exact". In other words: this is just maths, folks, nothing is really happening here, move along please.

Bluetooth epidemics on paper Bluetooth is a short range short-wave radio communication protocol which was designed as an alternative to traditional infrared communication (e.g. IrDA) in order to create small range "Personal Area Networks" of mobile devices. An important improvement over IrDA is the lack of the requirement of line of sight among devices, which incidentally makes it also useful as a malware propagation vector or attack target, since it allows for "casual" or unwanted interaction.

Even if Bluetooth is theoretically quite robust, since late 2003, a number of security issues in various specific implementations of the standard stack surfaced. Such attacks are very well described on the website [9], and they allow different degrees of data access (from the agenda to any file on a vulnerable device), communication interception, up to and including running any AT command taking full control of the phone, something that can be effectively used to transform a telephone into a spyphone [10]. To further stress that implementation glitches lurk below the surface, on June 2008 there was a very interesting security bulletin from Microsoft [11] which reported a vulnerability in the Bluetooth stack in Windows that could allow remote code execution, with system privileges. Most of these attacks can be ran from cellphones or portable devices, or can be ran from a distance using long range antennas and modified Bluetooth dongles (up to ranges of the order of 1 mile).

These flaws demonstrate how, in many cases, it is possible to steal information from mobile devices, controlling them from a distance, making calls, sending messages, or even connecting to the Internet. This type of problems is traditionally handled, in computer systems, with the release and application of patches. However, this approach does not extend to GSM handsets, since in most cases a firmware update can be performed only at service points and shops, not by the customers themselves: therefore many vulnerable phones and firmwares keep going around even long after a vulnerability is discovered.

Viruses for mobile devices propagating over Bluetooth also reportedly exist. The propagation of a Bluetooth virus can take place in several different ways. The most common, until now, is through simple *social engineering*. The worm sends messages with copies of himself to any device which comes into range through an OBEX push connection (OBEX is the protocol used for exchanging binary objects over Bluetooth). There are different profiles for this service, and "push" is the profile generally used for phone to phone occasional transfers without authentication (e.g. for exchanging electronic business cards). Much like in the case of e-mail worms and trojans, the receiver, finding an "attractive" message on the cellular phone with the invitation to download and install an unknown program, often has no clue that this can pose a danger. For instance, Cabir [12], one of the first cellular phone worms, and the first case of malware

able to replicate itself only through Bluetooth, used this technique. Using some vulnerabilities [13], also seemingly innocent files such as images could be used as a viral propagation vectors. Bluetooth attacks, such as the ones described above, could also be used: but since they are quite platform-specific, they are a difficult and unreliable mean of propagation when compared to the simplicity of social engineering.

A number of models have been proposed for Bluetooth worm propagation, almost invariably showing great propagation potentials [1, 14–16]. Antivirus vendors also claimed every year to be the year of mobile malware, but such predictions constantly failed to materialize [17].

In order to assess the effective prevalence of such targets, we are building a set of Bluetooth honeypots named BlueBat [18]. The name is a joke on the broader research project, WOMBAT (Worldwide Observatory of Malicious Behaviors and Attack Threats) of which they are a part. WOMBAT is an European research project which aims at providing new means to understand the existing and emerging threats that are targeting the Internet economy and the net citizens [19].

This type of sensor was already proposed by several authors (notably in [20], but without code release or results), but never made publicly available and/or deployed on a wide scale (the only such experience being briefly reported in [21], without actually publishing results and in a setup more similar to our former experiments in [1]). Our aim, on the contrary, was to develop a practical approach, in which low cost, inexpensive sensors could be realistically deployed to gather information on a wide and diverse population.

Our preliminary results are threefold. First of all, we positively confirmed what we already demonstrated in [1]: a lot of Bluetooth devices are kept enabled and in visible mode, paving the way for potential attacks. Also, a relatively high percentage of users (up to 8%) will carelessly accept files over Bluetooth from unknown sources. This should spell trouble. However, during our first tests, we received only a limited number of files transmitted to the honeypots, only one of which potentially harmful but incorrectly transmitted. So, even if there are anecdotical tales of Bluetooth infection, the threat seems to be of limited diffusion as of today.

The explanations that come to mind are the difficulty of writing effective code which works across different mobile platforms and makes use of Bluetooth, even in the case of a benign application. This creates also a non-uniform population in which it is difficult to envision a common bug paving the way for automated worm transmission. In second place, from our tests, "casual" transmission of a file is quite difficult: a simple scan for devices takes seconds or even minutes, and then transmission happens, for each device, over several tens of seconds. During this lapse of time, shifting positions may very well place the target device out of range. This was actually predicted in [22], which went against the common perception that mobility helped spreading such worms [15]. Also, since Bluetooth is transmitted on a 2.4GHz band, which is absorbed by water, the human body itself acts as a shield and can easily interrupt transmission. Therefore, in the wild transmission of worms can effectively happen

only in a semi-static scenario. These preliminary results cast many doubts on the viability of Bluetooth as a worm propagation mechanism, and on the effectiveness of former Bluetooth spreading models, including our own [1]. This creates a need for updated models of viral propagation, and for a re-evaluation of the infection likelihood even in closed environments, which we are currently working on.

Conclusions Wireless and mobile security, and in particular worm propagation over wireless networks, is an interesting and novel concept. It challenges and thrills us, creating appealing newspaper titles in the way. However, we must ensure to check our models against reality (even if, for some scientific communities, this is evidently not particularly important...), and after predicting threats that failed to materialize we must be able to go back and understand where we went wrong. We are very, very likely to see an increasing number of wireless attacks.

Targeted penetration of wireless networks, or reflected attacks brought through roaming clients, will surely happen, and the "wildfire" worm scenario may very well materialize in the future. On the other hand, router-to-router attacks are not going to happen anytime soon, no matter how appealing they look on paper (in particular if the wrong parameters are chosen for the simulations). Not caring about "whether a given model does or doesn't reflect the real world" is a serious issue for anybody involved in security choices, and engineers should therefore take with caution any result which comes out of models grounded on thin air. Otherwise, we might end up deploying anti-virus software on wireless routers, as opposed to doing something more sensible.

Speaking of anti-virus software deployed to respond to unlikely threats, Bluetooth worms are not yet here (not in a raging fury, at least), and in spite of all our models they do not seem likely to come. Bluetooth is just too unreliable to give birth to a real pandemic, unless something major changes in range, stability of communication, and most importantly unless a way to reliably write portable applications comes out. On the other hand, targeted attacks on high-profile devices are probably happening and will keep happening below our radar, because anti-virus software is not designed to deal with them, but only with self-propagating malcode that has not yet left the zoo.

Performing risk assessment is still in many ways an art, rather than a science. But even the most skilled Gypsy will have troubles reading the future in a stained crystal sphere: it is high time to review the mathematical model we use and ensure that they reflect reality.

References

[1] Luca Carettoni, Claudio Merloni, and Stefano Zanero. Studying bluetooth malware propagation: The bluebag project. *IEEE Security and Privacy*, 5(2):17–25, 2007.

[2] A. Tsow, M. Jakobsson, L. Yang, and S. Wetzel. Warkitting: the drive-by subversion of wireless home routers. *Journal of Digital Forensic Practice*, 1(3):179–192, 2006.

[3] P. Akritidis, W. Y. Chin, V. T. Lam, S. Sidiroglou, and K. G. Anagnostakis. Proximity breeds danger: emerging threats in metro-area wireless networks. In *SS'07: Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*, pages 1–16, Berkeley, CA, USA, 2007. USENIX Association.

[4] Hao Hu, Steven Myers, Vittoria Colizza, and Alessandro Vespignani. WiFi networks and malware epidemiology. *Proceedings of the National Academy of Sciences*, 106(5):1318–1323, 2009.

[5] S. Stamm, Z. Ramzan, and M. Jakobsson. Drive-by pharming. *LECTURE NOTES IN COMPUTER SCIENCE*, 4861:495, 2007.

[6] Zlob trojan. available online, `http://en.wikipedia.org/wiki/Zlob_trojan`, retrieved in 2009.

[7] Bluepill. available online, `http://dronebl.org/blog/8`, retrieved 2009.

[8] E. Tews, R.P. Weinmann, and A. Pyshkin. Breaking 104 bit WEP in less than 60 seconds. *Lecture Notes in Computer Science*, 4867:188, 2007.

[9] Trifinite.org website. `http://www.trifinite.org`.

[10] Pierre Betouin. Dossier sécurité bluetooth - partie 5 - scénarios d'attaques & synthése. Available online at `http://www.secuobs.com/news/05022006-bluetooth5.shtml`.

[11] Microsoft security bulletin nr. 30, 2008. Available online at `http://www.microsoft.com/technet/security/Bulletin/MS08-030.mspx`.

[12] Cabir. Analysis available online at `http://www.symantec.com/security_response/writeup.jsp?docid=2004-061419-4412-99`.

[13] Motorola RAZR JPG Processing Stack Overflow Vulnerability. `http://www.zerodayinitiative.com/advisories/ZDI-08-033/`.

[14] Jing Su, Kelvin K. W. Chan, Andrew G. Miklas, Kenneth Po, Ali Akhavan, Stefan Saroiu, Eyal de Lara, and Ashvin Goel. A preliminary investigation of worm infections in a bluetooth environment. In *WORM '06: Proceedings of the 4th ACM workshop on Recurring malcode*, pages 9–16, New York, NY, USA, 2006. ACM.

[15] James W. Mickens and Brian D. Noble. Modeling epidemic spreading in mobile environments. In *WiSe '05: Proceedings of the 4th ACM workshop on Wireless security*, pages 77–86, New York, NY, USA, 2005. ACM.

[16] Guanhua Yan and Stephan Eidenbenz. Modeling propagation dynamics of bluetooth worms (extended version). *IEEE Transactions on Mobile Computing*, 8(3):353–368, 2009.

[17] G. Lawton. Is it finally time to worry about mobile malware? *Computer*, 41(5):12–14, May 2008.

[18] A. Galante, A. Kokos, and S. Zanero. Bluebat: Towards practical bluetooth honeypots. In *2009 IEEE International Conference on Communications*, Dresden, Germany, June 2009.

[19] WOMBAT project website. `http://www.wombat-project.eu/`.

[20] Adam Laurie, Marcel Holtmann, and Martin Herfurt. Bluetooth Hacking: Full Disclosure. `http://trifinite.org/Downloads/syscan2005_slides.pdf`, 2005.

[21] Post on F-Secure's blog. `http://www.f-secure.com/weblog/archives/00000836.html`.

[22] Guanhua Yan and Stephan Eidenbenz. Bluetooth worms: Models, dynamics, and defense implications. In *ACSAC '06: Proceedings of the 22nd Annual Computer Security Applications Conference*, pages 245–256, Washington, DC, USA, 2006. IEEE Computer Society.