



White Paper

South : Station

Aaron Greenspan

Date: January 31, 2005
Topic Area: Security

Wireless internet access has become a pervasive phenomenon in America's cities today, and there are many reasons why that is a good thing. Almost anywhere you go, whether it is a small coffee shop, or a car dealership, or an airport, or even the middle of a sidewalk, there's a good chance you'll be able to find a wireless signal, obtain an IP address, and start using the internet.

As I'm writing this paper from my chair near the corner of my office in Boston's Financial District, there are six wireless networks available for my laptop computer to sign onto, two of which require no encryption whatsoever. None of them belong to my company or myself personally. One of them does belong to a company I know to be nearby, and should I choose to sign onto its network, I have full access to files on their Windows NT and Macintosh servers. Sometimes, I take this action without my even knowing it; for some reason, even though I've asked it not to, Microsoft Windows XP occasionally opts for the best wireless connection instead of my wired ethernet cable, which is faster. When this occurs, I am able to browse the inter-

net using the nearby company's DSL line (for which they are presumably footing the bill), but I usually cannot tell the difference.

It has already been well-documented that wireless routers intended for home use are often insecure due to the fact that hapless customers tend to leave their default settings as they are. This usually means that you can sign into any home router with relatively obvious authentication information, such as the username "admin" and the password "admin." This is not always the case, of course. Depending on the manufacturer and model, the password might throw you off (some use "1234"), but it is

never very hard to figure out. If for some reason you cannot guess it, a simple search on the internet for "default router passwords" will reveal a default password for every router you ever might want to know about. These pages sometimes follow the basic syntax for authentication information, which involves the username, followed by a colon, and then the password. Decoding the information is not difficult. All that's left to do for the visitor of such a page is match up the model number on the router with the one on

What is truly worrisome is what might happen if similar security issues with wireless routers really began to affect our businesses, financial institutions and our physical infrastructure: the basic framework of our society.



his or her screen.

The damage that can be done in this fashion is usually underestimated, for hacking often assumes the form of a chain reaction, as you will see in this paper. In other words, each time a hacker finds a password, it only makes it easier to find the next one.

Once a hacker knows the password for a router, firewalls can be shut down. When those are down, ports are open, and viruses can infiltrate networks easily. Viruses often bring with them “malware:” spyware, keystroke loggers, data loss, and a plethora of other technical problems. Based on observations at Think, almost all Windows-based desktop computers in use today are afflicted by at least one of the aforementioned problems. An incredible amount of the spam we receive in our inboxes comes from our nextdoor neighbors, who do not even know that they are sending it. Misconfigured routers are somewhere along the beginning of the chain.

It is worrisome to think what might happen if these kinds of security issues really began to affect our businesses, financial institutions and our physical infrastructure: the basic framework of our society. It is worrisome only because it is already happening.

South Station is a major transportation hub in downtown Boston, Massachusetts. It serves thousands of passengers and commuters each day, who travel by rail on Amtrak, the Massachusetts Bay Transportation Authority (MBTA) Commuter Rail, by subway on the MBTA Red Line, by bus on the MBTA Silver Line, and by bus on several commercial coaches. It is located across the street from One Financial Center, a skyscraper more than thirty stories high; next to Fort Point Station, Boston’s main Post Office; and is still undergoing construction as a result of the “Big Dig,” the multi-billion dollar construction effort that

has overtaken Boston for more than the past decade.

One might think that such a crucial cog in the machine of a major city would simply be well-protected by default, especially in this day and age when federal and state government agencies have done all they can to make the public hyper-aware of security issues. Yet, as it turns out, South Station suffers from the same problems as the Joneses.

South Station provides wireless internet access to travelers as a fee-based service. If your train to Washington, D.C. happens to be late, you can bide your time by logging onto the internet to check your e-mail for \$6.95 per hour. Or, if you prefer, you can buy a daily or monthly pass for slightly more. Whenever you try to visit a web site without paying for the service, you are automatically redirected back to a special home page that South Station has set up for the purpose of making you pay.

The billing and technical operations of this system are handled by a company called Atlantis Technology Corporation, located in Maynard, Massachusetts. Atlantis operates a service called guestBOX that provides everything a major transportation hub such as South Station needs to offer wireless internet access to customers.

The customized South Station home page that encourages wireless to “sign up now” also displays a nice photograph of the Station at the top of the screen, if only to make the notion of paying for access aesthetically pleasing. As it so happens, this single photograph represents the only crack in the wall of security necessary to take control of the entire building’s wireless system, as well as at least thirty others in the state of Massachusetts.

On the World Wide Web, photographs are stored in files separate from web pages them-



selves. Anyone can right click on a picture, choose to view the picture alone, and see the address where it is stored. Of course, it is always possible to store more than one file in a single directory. Removing the name of the picture's file, but not the directory that it is stored in, from the address bar in the web browser should not reveal any additional information on a properly-configured server. On an improperly-configured server, there is no telling what could happen.

Atlantis apparently forgot to configure its guestBOX server to prevent people from seeing the contents of directories, because it displayed the names, types and file sizes of every image in the South Station image directory—not just the one picture on the home page. The fact that such a trick worked in one directory meant that it probably worked for them all.

Indeed, it did. While South Station had been in the “images” folder within directory “26,” it appeared that guestBOX had other clients, and their sites' images were stored in directories labeled with other numbers. Simply substituting the “26” with a “25” displayed the images of a different client, and then another client with “24,” and another with “23,” and so on.

The server offered much more than images, however. Requesting the complete contents of the “26” directory itself resulted in an error message that revealed more information still:

```
Warning: main(WEB_PATH/client_pages//header.php): failed to open stream: No such file or directory in /usr/local/guestBOX/www/client_pages/25/index.php on line 3
```

```
Fatal error: main(): Failed opening required 'WEB_PATH/client_pages//header.php' (include_path='.:usr/local/lib/php:usr/lib') in /usr/local/guestBOX/www/client_pages/25/index.php on line 3
```

Now it was apparent that there was a file called “header.php,” apparently missing due to a programming error, somewhere in the directory. Adding “header.php” to the location in the address bar indeed brought up a page that should have contained the aforementioned images, but did not, due to yet another mishap somewhere in the guestBOX source code. An easy guess revealed that there was also a file called “footer.php” in the same directory.

The HTML source code for header.php did mention a folder called “common.” Typically, programmers use directory names such as these to consolidate files that are used by more than one part of a program or site into a single, centralized location. It did not take long to find the “common” directory, and display the list of files there, as well. Some of their names, such as “auth_user.php,” implied that they interacted with a credit card system, and bringing up one of the files confirmed this suspicion: it displayed a form asking for a username and password prior to credit card authentication.

At this point it was already apparent that the programmers at guestBOX had not done their homework with regard to security. A properly designed system would not have used sequential numbers for each client without some sort of authentication mechanism.

It was still not possible to access the internet without paying a fee to guestBOX, except for one site: <http://www.guestbox.com>. On the bottom left corner of the official guestBOX home page, a small form asked for a username and password for “client login.” The classic `admin:admin` combination failed to work. However, the next guess did. Using `test:test` (a favorite of programmers debugging their own work before it is ready for use) brought up the client administration web site for one of guestBOX's other cli-



ents, a restaurant I had frequented often myself: The Wrap.

In the administrative area, it was possible to set prices for wireless access, change the payment mode from “free” to “password” or “credit card” and back again, view security violations (the site reported that there were none at the present time), view revenue reports, change The Wrap’s custom home page, buy additional guestBOX hardware products, or contact support. It was amazing that a user outside of one of The Wrap’s physical locations could sign in at all, let alone just ten minutes after viewing the “secure” South Station home page, which gave no indication of any connection between the two establishments.

The guestBOX main site was still active. Another educated guess worked: the secret authentication key for Boston’s venerable South Station was, in fact, `south:station`.

Now I had access to all of the same features as I had for The Wrap, except that they applied to a train station, not a sandwich shop. Not that it was necessary to be in the train station. Anyone with internet access from Montana to Tokyo capable of performing similar guesswork could have effected the exact same changes remotely. Nevertheless, I was inside South Station with a fairly common, inexpensive laptop computer, in public, in plain view of a man sipping a latte at the table behind me, the Amtrak officials working on the second floor, and the MBTA Police, should they have decided to come and inspect my screen.

To ensure that the “administration site” I had stumbled upon was not merely an inactive test site, I changed South Station’s payment mode from “credit card” to “free.” A quick visit to the web site of The New York Times, which had not worked previously, confirmed that there were still worries about the vote in Iraq. I immediately

switched back South Station’s payment method to “credit card,” and even though I could still surf the internet (since my computer’s wireless MAC address had essentially been grandfathered into the system), another visit to the address of the guestBOX / South Station custom home page ensured that they were charging for access once more. On some level, the guestBOX system did work.

In fact, it worked fairly well. I could see the revenue figures for South Station’s wireless internet activity since September, 2004. With a little bit more time and effort, it is not extreme to surmise that it would have been possible to find the location of the MySQL database on guestBOX’s server’s hard drives where credit card numbers were stored so that they could be charged again, week after week.

South Station is by far not the only vulnerable part of our nation’s transportation infrastructure. Terminal C in the Cincinnati airport offers wireless internet access provided by Cincinnati Bell. By default its routers direct your web browser to a custom home page similar to the one at South Station, though designed for the airport, at <http://192.168.4.100>. Changing the address by one number to <http://192.168.4.101> reveals the name, username, and password of every Cincinnati Bell wireless customer who has ever walked through that terminal, and as an added bonus, you can add more time to anyone’s account you choose. It also offers up the e-mail address of the system’s programmer, and some directory listings that are so misconfigured that the files listed do not even display properly. One of them even displayed the root password—the holy grail of hacking—for one of Cincinnati Bell’s other servers: “cisco.” (Cisco Systems, Inc., of course, is the once high-flying dot-com darling that still designs high-capacity routers.) For those



uninterested in tapping into the nation's phone system, however, it is easiest to simply check your e-mail with the dummy `wifi:wifi` account.

Is it worth worrying about people "saving" \$6.95 per hour by essentially stealing free internet access? In the long run, the answer is no. It is conceivable that there might come a day when charging for internet access is akin to charging for tap water at a restaurant. There are five far more troubling aspects of the problems described in this paper.

The most obvious is that customers currently pay for these services with credit cards. If programmers are so lazy as to set high-level passwords such as "station" and "cisco," it is not hard to imagine that they are also neglecting to take adequate precautions to protect our credit card data.

A more farfetched, but very real possibility, is that computers or workers at airports and train stations *also* use these same networks to make everything tick. If that is the case, it might be possible for an intelligent high school student to start changing train timetables or rerouting baggage.

Thirdly, guestBOX's programming errors permitted a full compilation of the company's clients, as long as I was interested in searching for them. I stopped counting at 29, but aside from South Station, hotels, restaurants, coffee shops, and car dealerships were all among the types of businesses affected by the problem. These companies were spread out all across Massachusetts, and I had control of their networks.

Furthermore, when I attempted to contact Cincinnati Bell about their grievous programming error, I received the cold shoulder. It was impossible to find the right person to talk to, and there was always the risk of incurring the wrath of a large organization's legal department. At

one point a customer service representative actually suggested contacting the company's "abuse" e-mail account to report myself. If reporting security problems does not get easier for so-called white hat hackers, then there is a very good chance that problems will not be reported until it is far too late.

Lastly, the overall design of today's wireless networks suffers from a certain vulnerability that malicious hackers will be sure to exploit, if they have not done so already: the lack of any ability to track users. When a user connects to a web site, such as the one run by guestBOX, the web server handling the request almost always takes note of the user's IP address, the referring link, the type of web browser being used, etc. On a conventional wired connection, such as a DSL or cable line, the IP address is somehow connected by an Internet Service Provider's database to the name of a real person paying for the account, meaning that there is at least accountability at some level. If, however, enforcement agencies want to learn the identity of a perpetrator who has broken into a system using a free wireless network, it is essentially a pointless endeavor, for every interaction is completely anonymous. IP addresses are handed out to anyone and everyone who requests them, without any sort of identification process beforehand, so there is no way to tell who is who. Unless something is done to force accountability for wireless devices, perhaps by recording ethernet MAC addresses (which are unique and hard-coded to a physical piece of hardware), the smartest hackers will use loopholes in our current systems to break into them, literally in plain sight. And when they're done, they will simply get up from their seats, as I did two days ago, and calmly walk away—maybe with your data, or maybe having planted the seeds for something far worse.