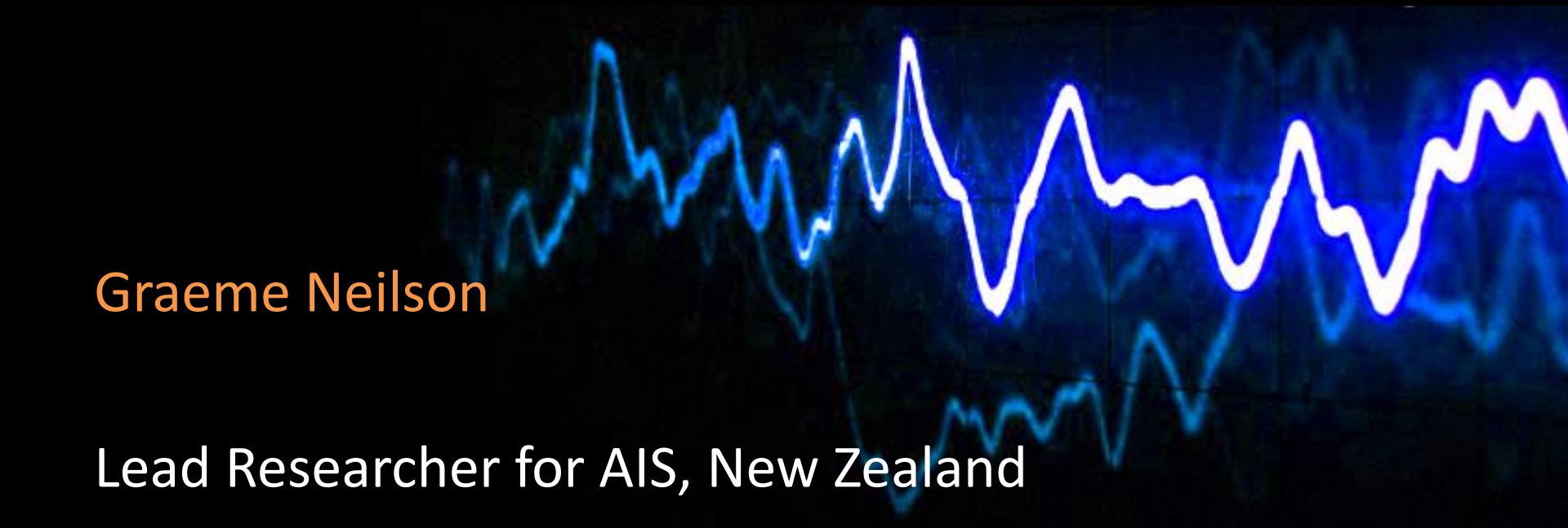
BLUEVOX

Attacking Audio One Time Passwords at 1100Hz

Troopers 13, Heidelberg, March 2013

Who



Graeme Neilson

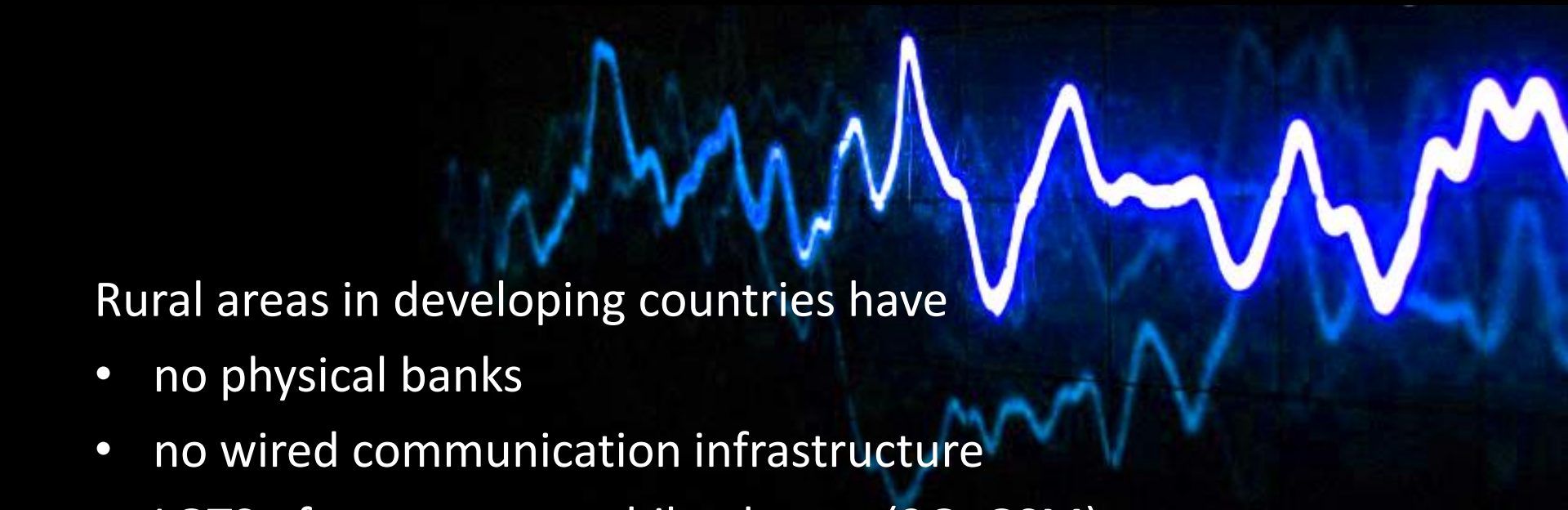
Lead Researcher for AIS, New Zealand

- Reverse Engineering, Cryptanalysis, Networking
- Troopers, H2HC, BlackHat, CanSecWest
- Firewall Rootkits, Cryptography

What

- Background of talk
- Use of Audio One Time Passwords on websites
- Methodology for collection and analysis
- Describe an attack

Background
Micro-Finance



Rural areas in developing countries have

- no physical banks
- no wired communication infrastructure
- LOTS of non-smart mobile phones (2G, GSM)

Companies are leveraging mobile phones

- as digital wallets
- for authentication

Background

Center of Mobile Revolution : Africa

- 65% Market penetration
- Better access, made cheaper for the consumer.
- Huge cultural impact.

"Africa is the Silicon Valley of banking. The future of banking is defined here. It's going to change the world."

Carol Realini, CEO, Obo Pay, Califonia, US

Background

What The World Saw…

Background

What Was Actually Happening…

Background
Mobiles for Authentication

- Retail devices “Mobile Payment Terminal”
- ATMs “SIM card present”
- Ecommerce Websites “Very Secure Authentication”

1. An audio file is generated by a server
2. Server calls client associated mobile
3. Audio is sent to client to play the audio.
4. The mobile listens and then transmits the audio to the server.

Website Authentication

Overview

1. Log onto website by entering mobile number and 4 digit PIN
2. The server tells the backend to call your mobile number
3. Client answers the call and holds handset near speakers
4. Browser plays audio using Flash browser plugin
5. Mobile transmits audio to backend server
6. Backend server compares audio sent and audio received

Website Authentication

Login

Login to your Mobipay account

Phone number:

Passcode:

Login

New Client? click here!

To Login You Must:

Step 1: Login

Enter your phone number and passcode.

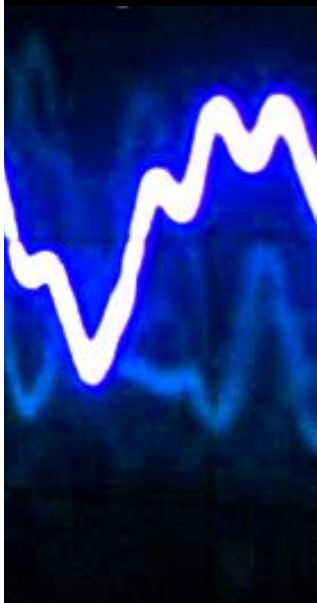
Make sure that your computer speakers are on

Step 2: Mobile Authentication

Mobipay will call you

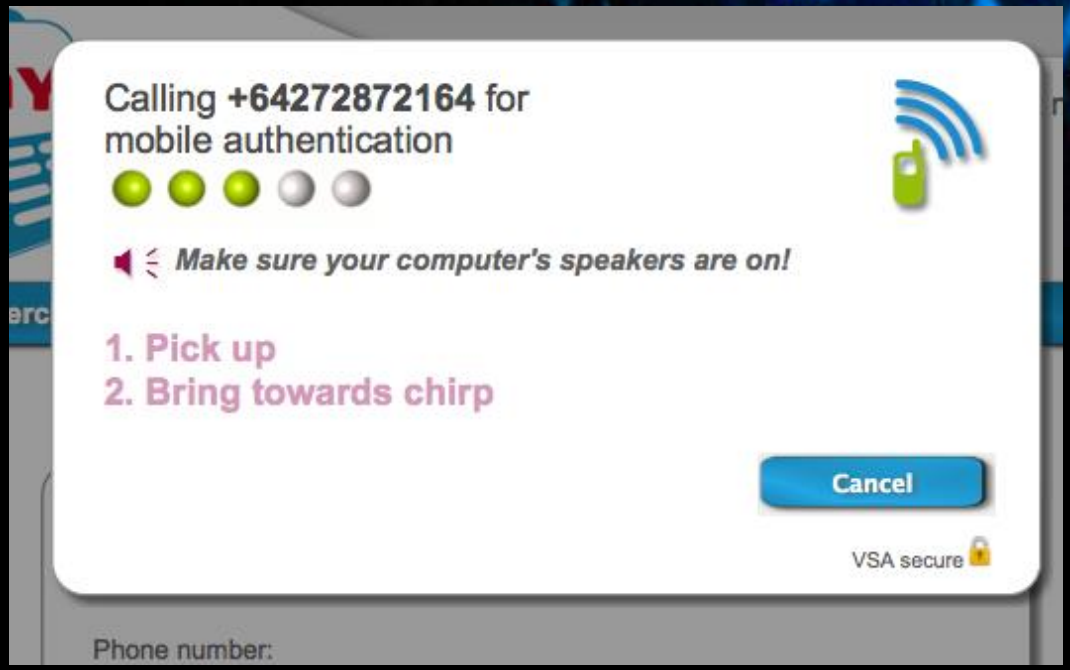
Answer the call and bring your phone towards the computer's speakers to complete your login.

If you do not have your mobile phone you can use your password.



Website Authentication

Audio One Time Password



Website Authentication Weaknesses

- Requires mobile numbers
  - Can enumerate phone numbers from login page
  - Can discover mobile blocks
  - ITU National numbering plans

  http://www.itu.int/oth/T0202.aspx?parent=T0202
- Require 4 digit PIN code
  - this is brute forceable on the website
- Authentication relies on the AOTP so if that is weak…

Audio One Time Passwords

Testing

- Are they random?
- Are they one time?
- Are they different for every backend?
- Are they different for every mobile number?
- Are there magic audio sequences?

Methodology

Overview

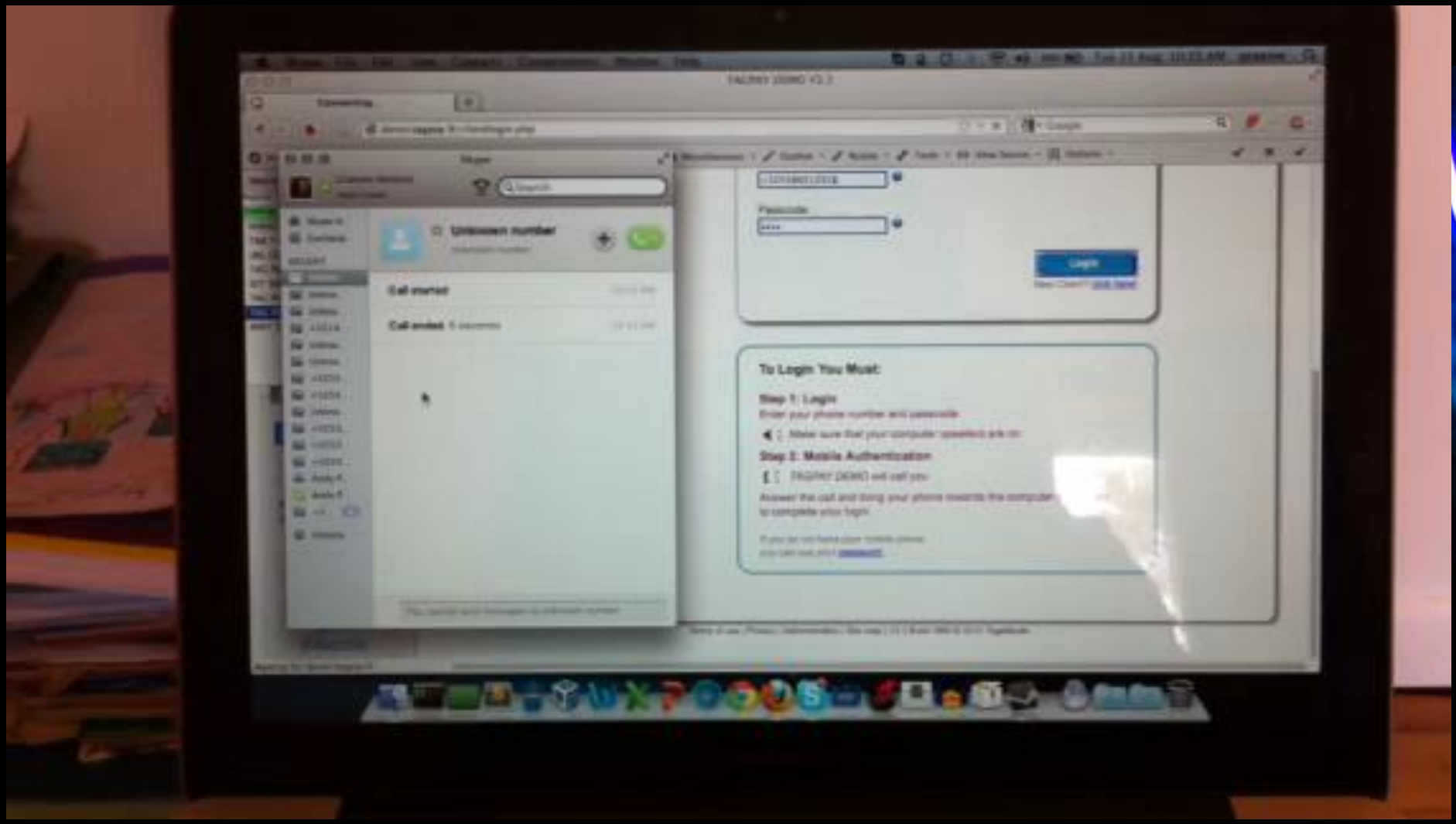
1. Collect a large sample of AOTPs
2. Extract the audio from the AOTP
3. Convert audio to a number sequence somehow…?
4. Analyse number sequence for randomness

Collection

mitmdump –p 1100 –w dump.out

- Register a Skype-In number
- Register Skype-In number with website
- Set Skype to auto-answer
- Set Firefox to use mitmdump HTTP proxy
- Use Firefox plugin iMacro to record a login session
- Set iMacro to loop 10,000 times and press play…

Collection


Unknown number
Call started
Call ended
To Login You Must:
Step 1: Login
Step 2: Mobile Authentication

Methodology

Overview

1. Collect a large sample of AOTPs
2. Extract the audio from the AOTP
3. Convert audio to a number sequence somehow…?
4. Analyse number sequence for randomness

Extraction Overview

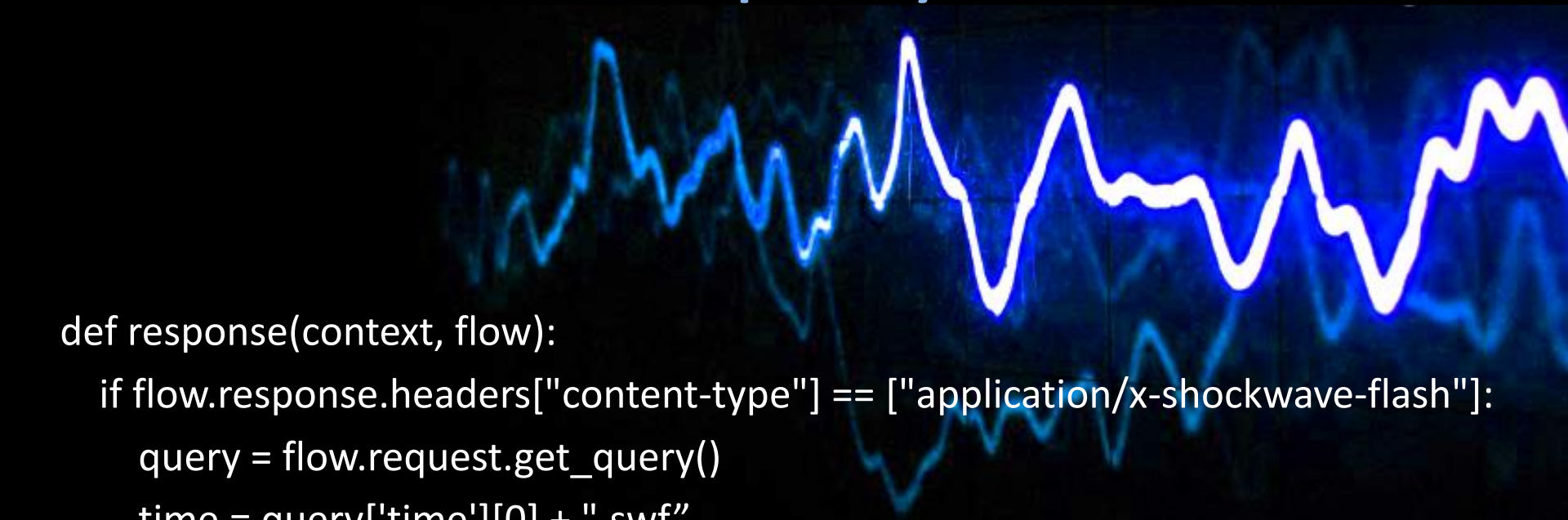
- Create mitmproxy python script to extract Flash files
- Use timestamp from client HTTP request as the file name

  mitmdump –s save_swf.py –r dump.out

- Use swfextract to extract mp3 from Flash

  swfextract –m file.swf -o file.mp3

Extraction
Flash from mitmproxy



```
def response(context, flow):
    if flow.response.headers["content-type"] == ["application/x-shockwave-flash"]:
        query = flow.request.get_query()
        time = query['time'][0] + ".swf”

        swf_file = open(time, "wb")
        swf_file.write(flow.response.content)
        swf_file.close
```

Extraction
Lifetime

- Compare each mp3 with the one before and remove identical mp3s

- Use the timestamps in the filename to determine lifetime of the AOTPs

Methodology

Overview

1. Collect a large sample of AOTPs
2. Extract the audio from the AOTP
3. Convert audio to a number sequence somehow…?
4. Analyse number sequence for randomness

Conversion Analysis

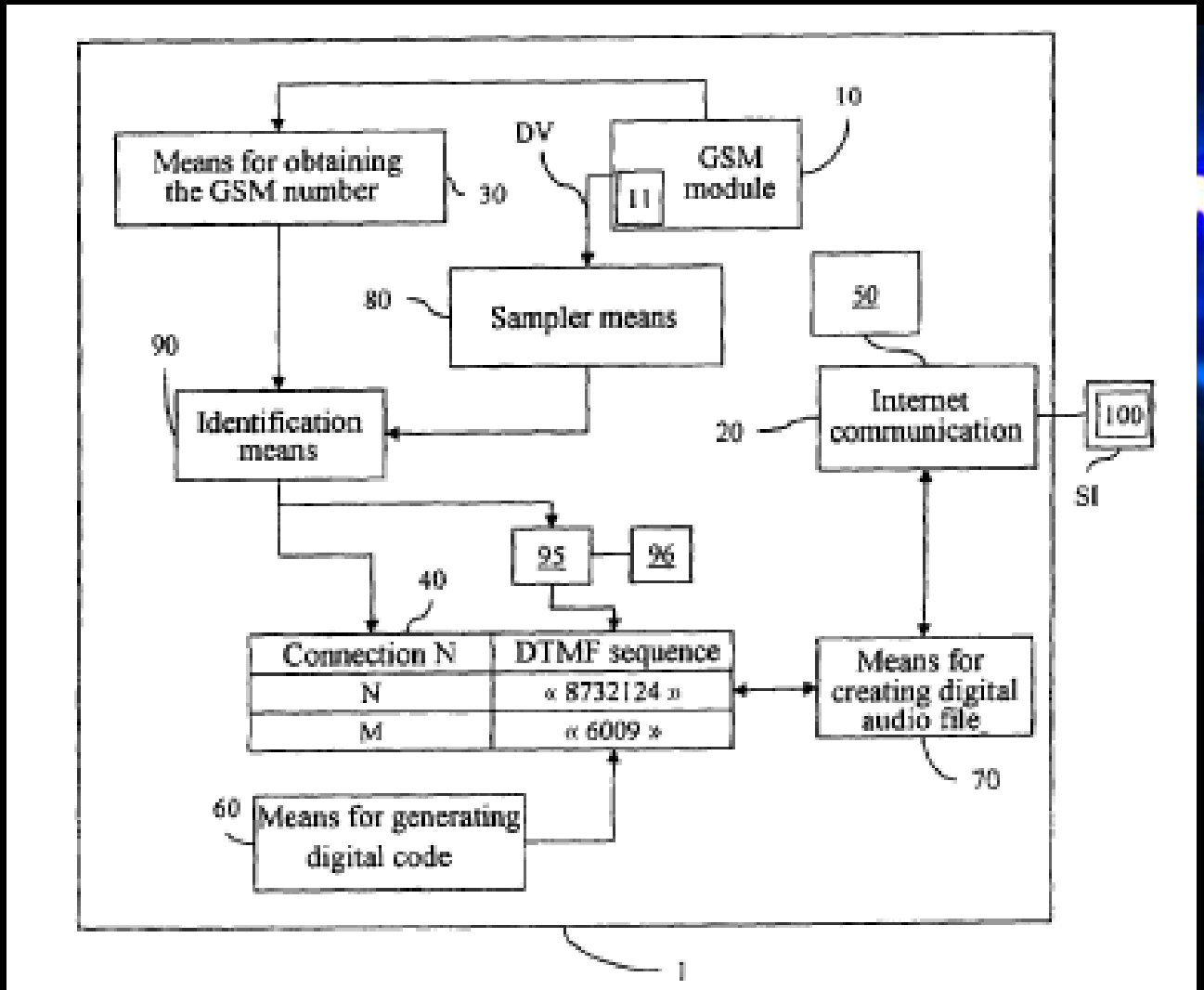
What they say:

- Patent says DTMF
- 250ms and contains 55 bits of information
- Signal processing engine verifies the integrity (CRC + FEC)

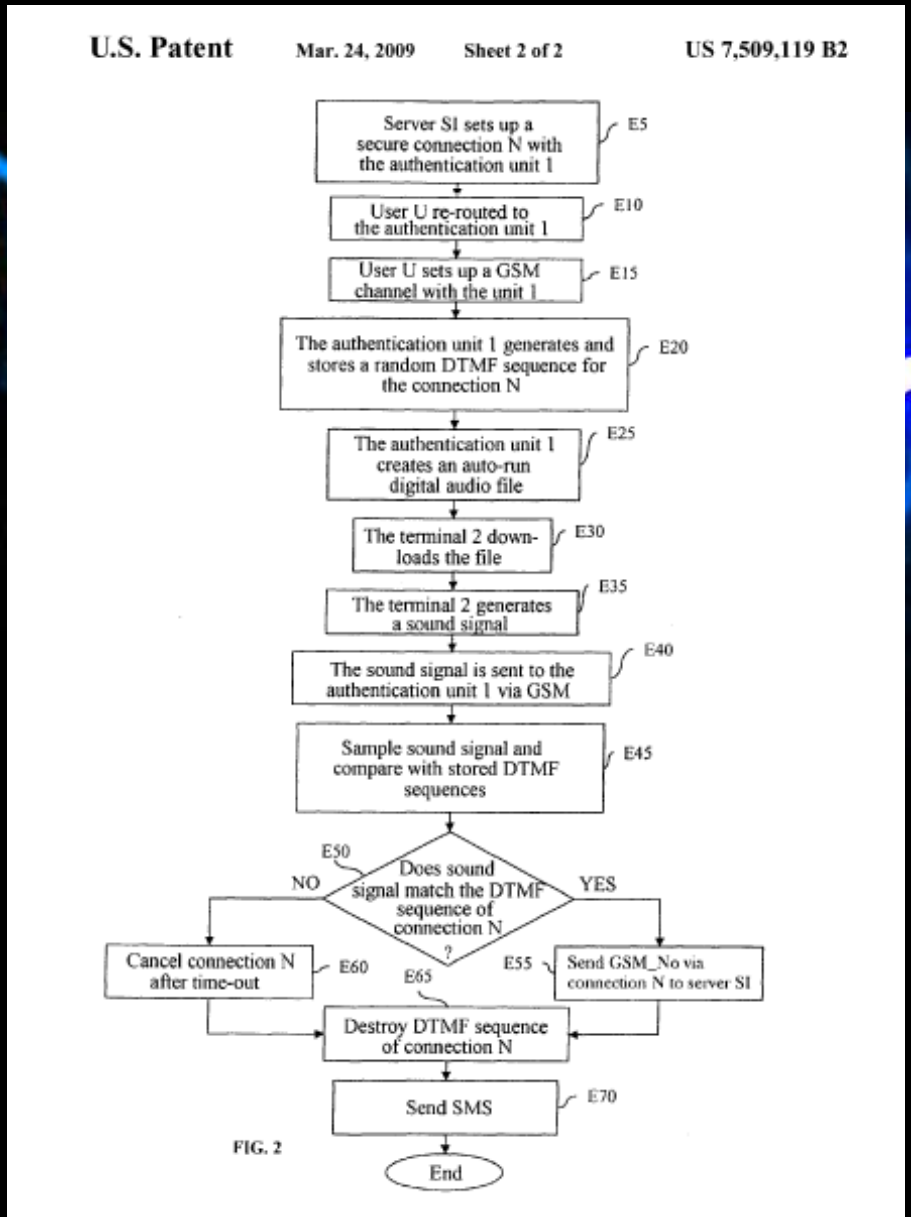
What it is:

- MF not DTMF
- AOTP is 1000ms
- Uses ‘2 out of 6 encoding’ to provide CRC and FEC

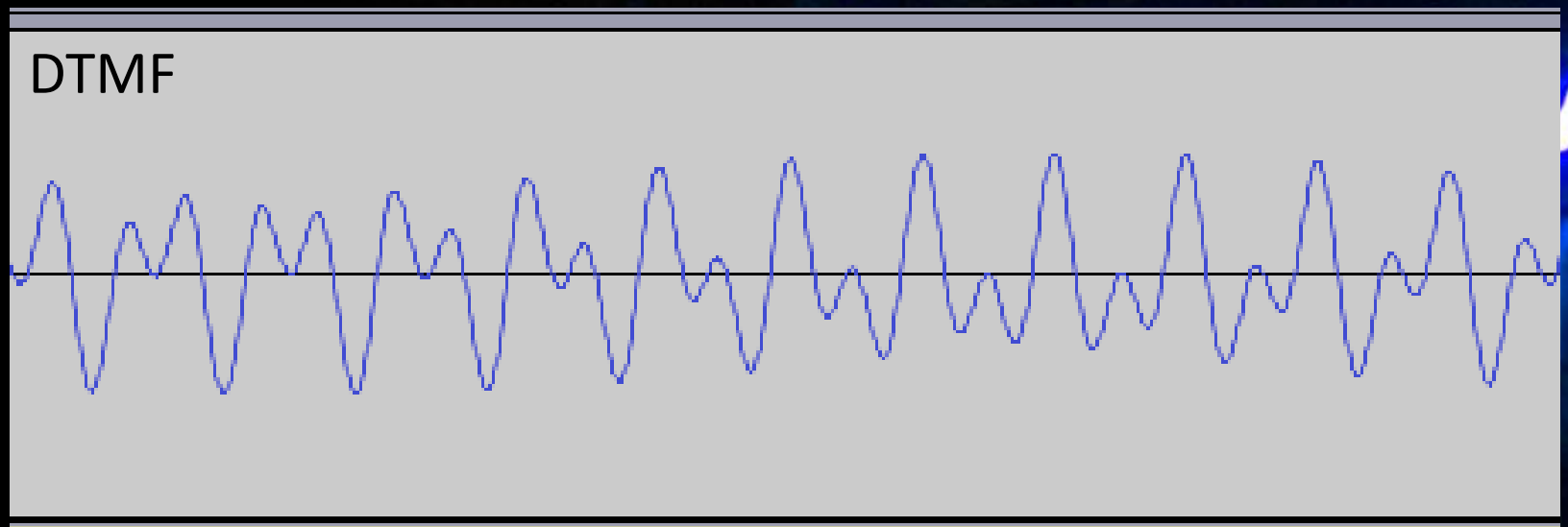
Conversion Patent

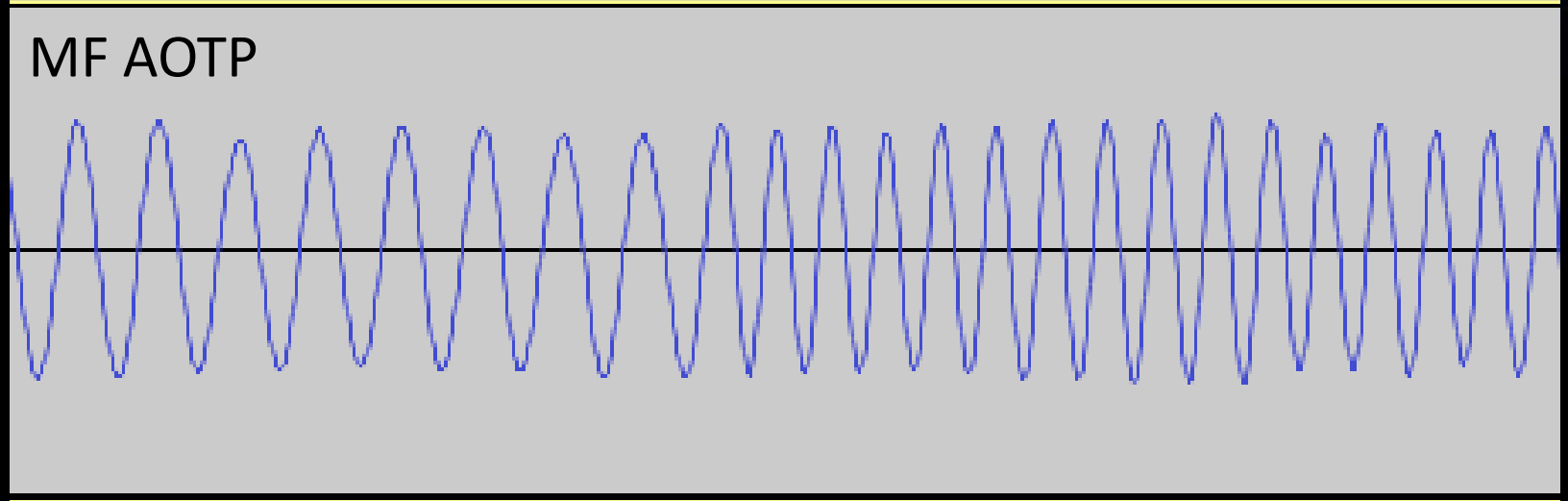


Conversion Patent



Conversion
MF not DTMF





Conversion
SS5 Protocol & In Band Signaling

- SS5 is a multi-frequency (MF) telephone signaling system in use from the 1970s for International Direct Distance Dialing

- Uses 2 in 6 encoding for register signaling

  700, 900, 1100, 1300, 1500, 1700 Hz

- 2 frequency (2VF) code used for line signaling

  2400, 2600 Hz

Conversion

SS5 2 in 6 Encoding

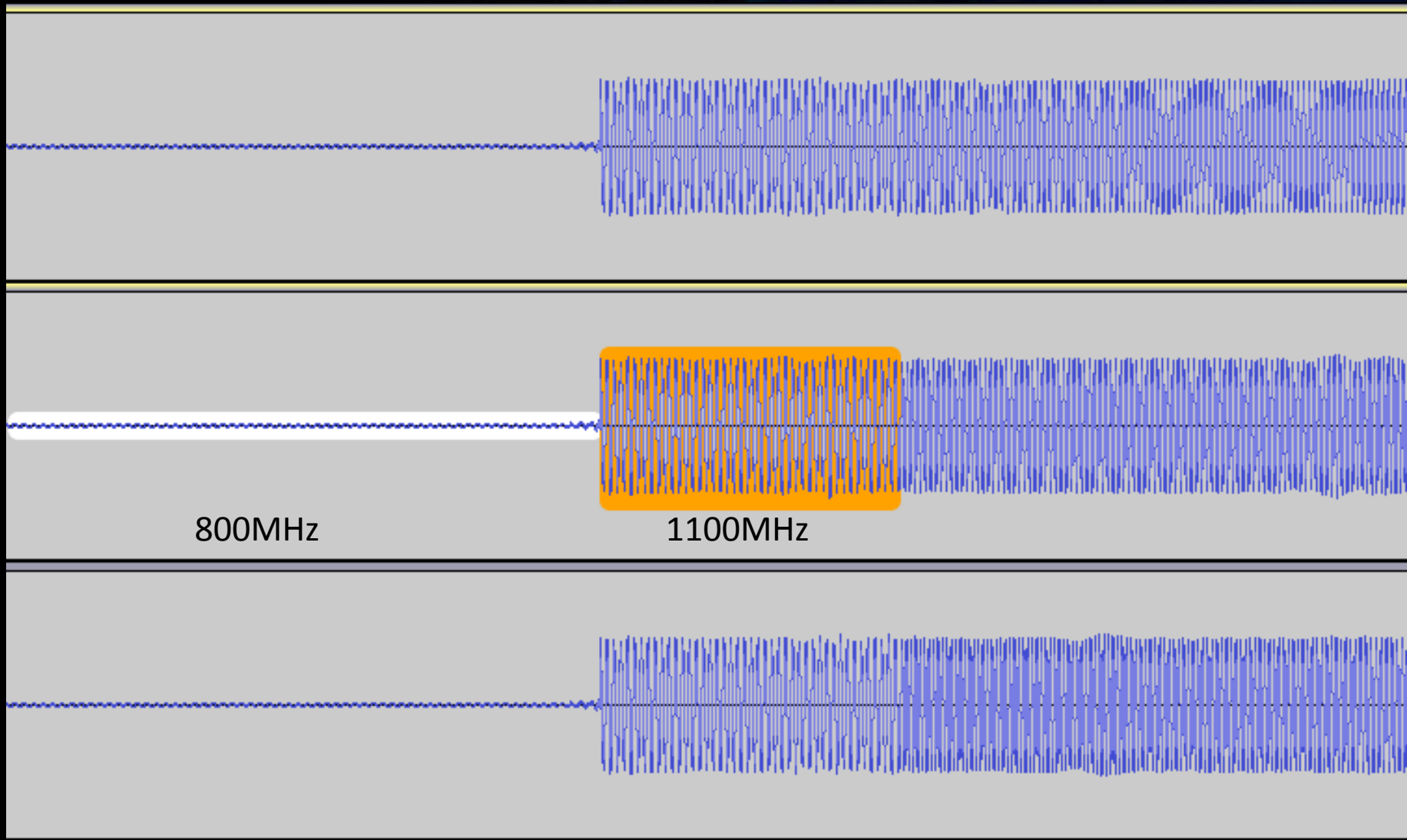
Frequency Pair	Digit
700 & 900	1
700 & 1100	2
900 & 1100	3
700 & 1300	4
900 & 1300	5
1100 & 1300	6
700 & 1500	7
900 & 1500	8
1100 & 1500	9
1300 & 1500	10

Frequency Pair	Code
700 & 1700	Code 11
900 & 1700	Code 12
1100 & 1700	Code 13
1300 & 1700	Code 14
1500 & 1700	Code 15

Conversion
In Band Signaling

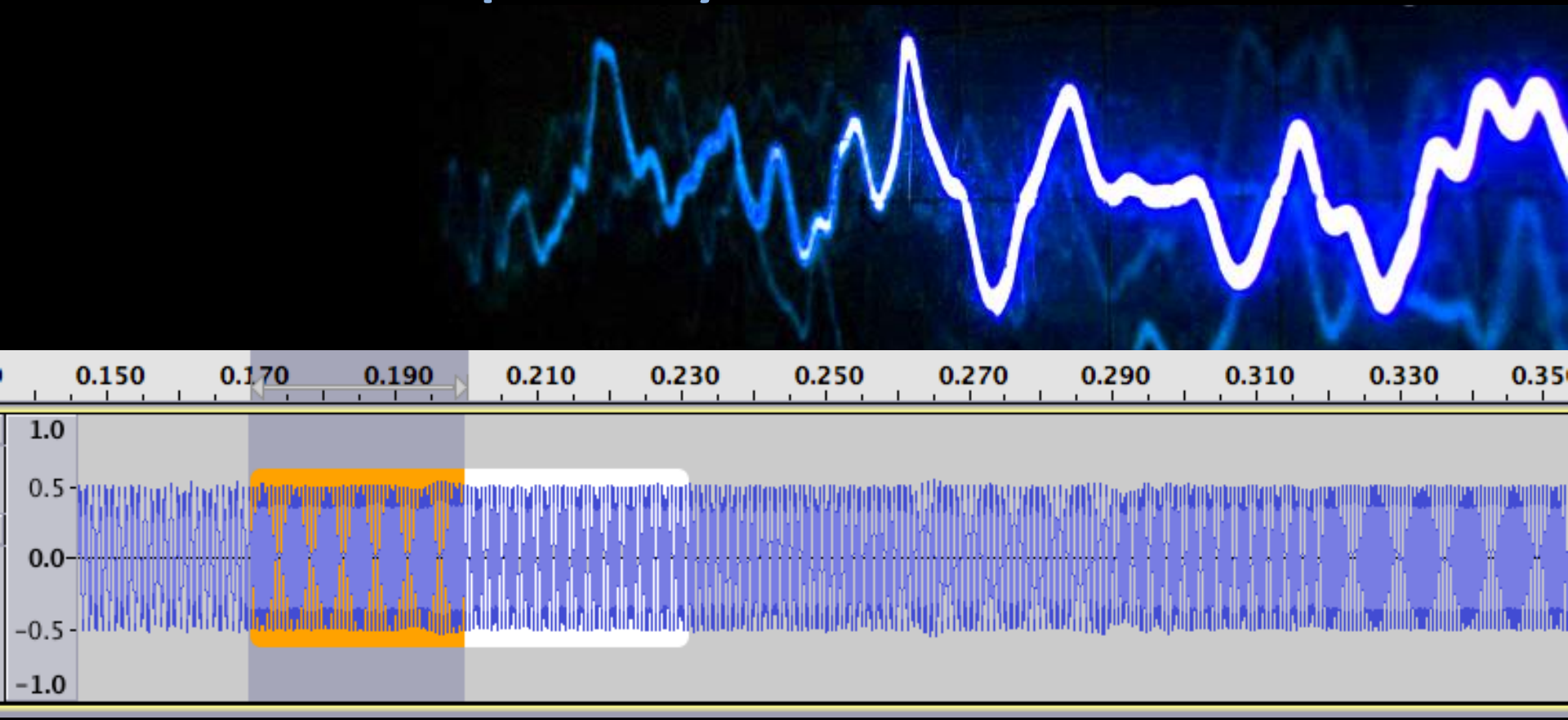
- Similarities to SS5
- Line signaling using 2 frequency (2VF) code:
  800 and 1100Hz
- Register signaling uses frequency pairs for 2 in 6 encoding: 1200, 1300, 1400, 1500, 1600, 1700 Hz

Conversion 1100Hz



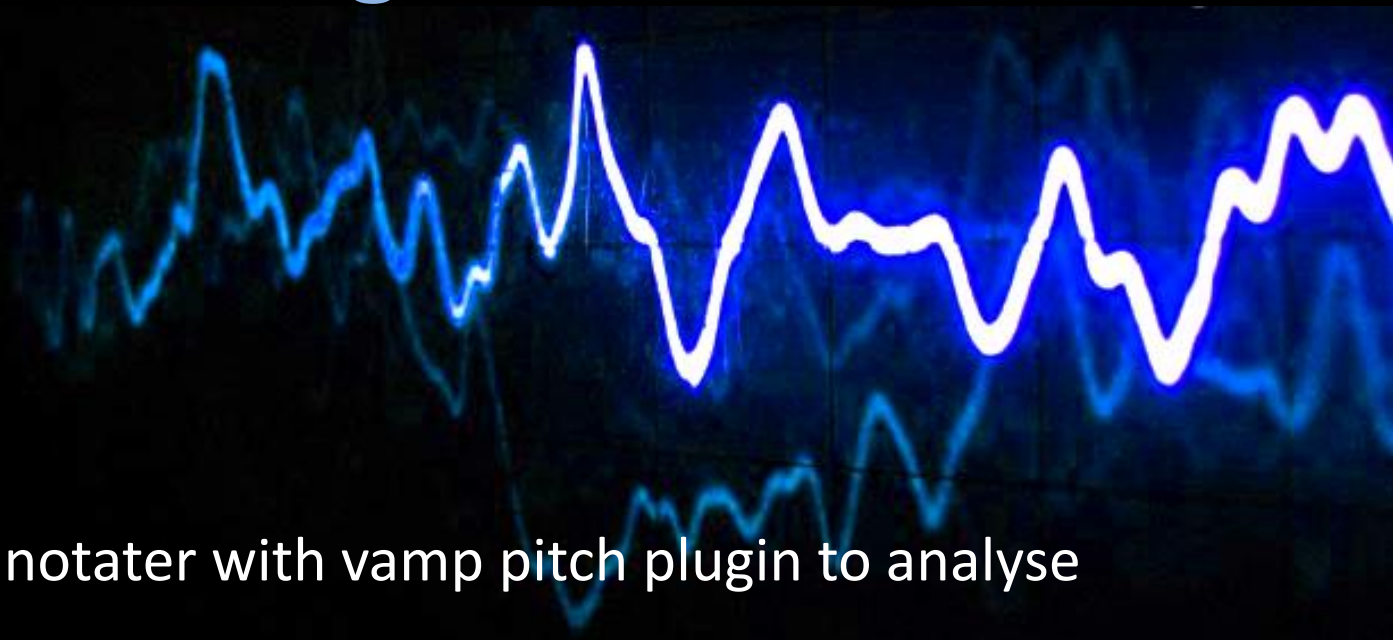
Conversion

Multi Frequency Pairs



Each pair is 2 frequencies of 30ms each

Conversion
Signal Processing



1. Use sonic-annotater with vamp pitch plugin to analyse frequencies:

```
sonic-annotater –t vamp-freq.n3 file.mp3 –w csv –csv-out > file.csv
```

2. Decode frequencies to an integer using 2 in 6 encoding

Methodology

Overview

1. Collect a large sample of AOTPs
2. Extract the audio from the AOTP
3. Convert audio to a number sequence somehow…?
4. Analyse number sequence for randomness

Analysis
Cryptanalysis

- Statistically analyse random numbers:
  - NIST STS http://csrc.nist.gov/groups/ST/toolkit/rng
  - TestU01 http://www.iro.umontreal.ca/~simardr/testu01/tu01.html
  - ENT http://www.fourmilab.ch/random/
- Correlate decoded AOTP values with the timestamp from the request?
- Predict AOTP?

Analysis
Disclosure

- Full disclosure would only harm users in developing countries
- Enough info in the slide deck

Attack
Phreaking

Need a tool for

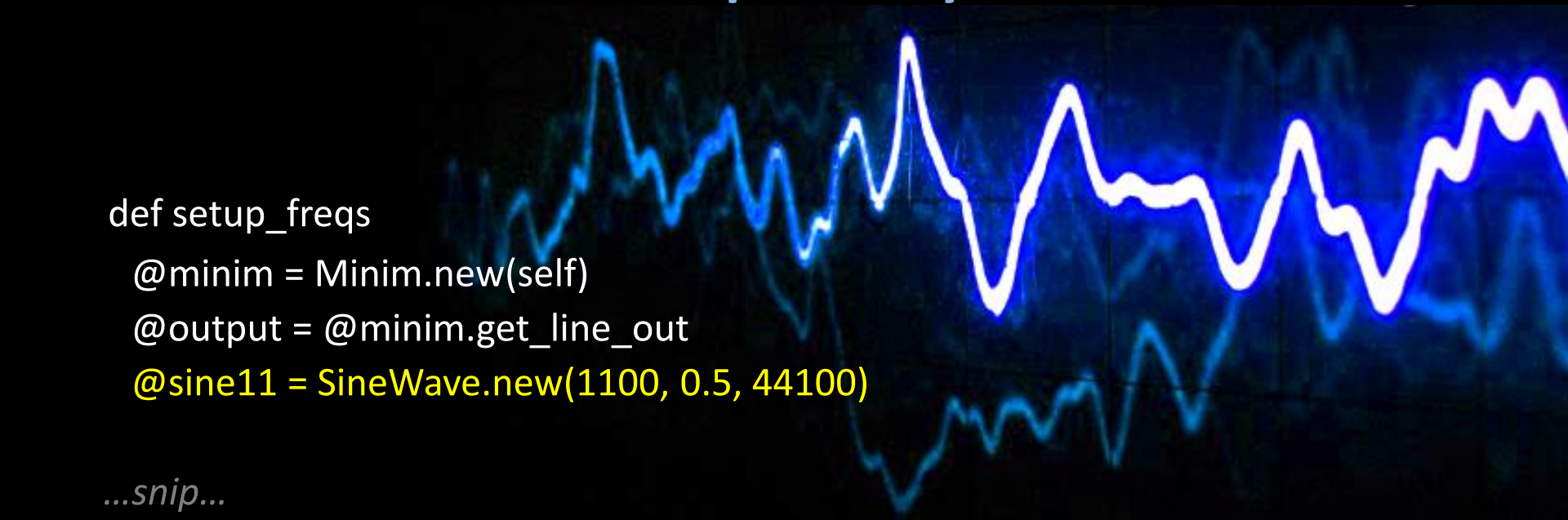
1. Generating predicted AOTPs

1. Phreaking the backend system when it calls to authenticate
   - Similar setup as collection
   - mute sound from Firefox flash plugin
   - run fuzzer so Skype sends output from phreaker to server

1. Requires ruby-processing, minim, real time kernel for msecs

Attack

MFP : Multi Frequency Phucker



```
 def setup_freqs
   @minim = Minim.new(self)
   @output = @minim.get_line_out
   @sine11 = SineWave.new(1100, 0.5, 44100)

 ...snip...
seq.each do |hexnum|
    freqs = two_in_six_encode(hexnum)
    @output.enable_signal(freqs[0])
    start = Time.now.usec
    while Time.now.usec < start + seq_length do nil end
    @output.disable_signal(freqs[0])
 ...snip...
```

BlueVoxing Attack?

How do we leverage AOTP weaknesses?

- Micro-finance means any attack must be a mass attack
- AOTP system assumes that calling the mobile number will reach the handset.
- We do NOT need physical access to the handset

BlueVoxing VoiceMail

- Record an AOTP as the voicemail greeting

- Ensure voicemail by
  - Calling the mobile twice in quick succession
  - Calling during the night

- Login -> server authenticates the AOTP voicemail greeting

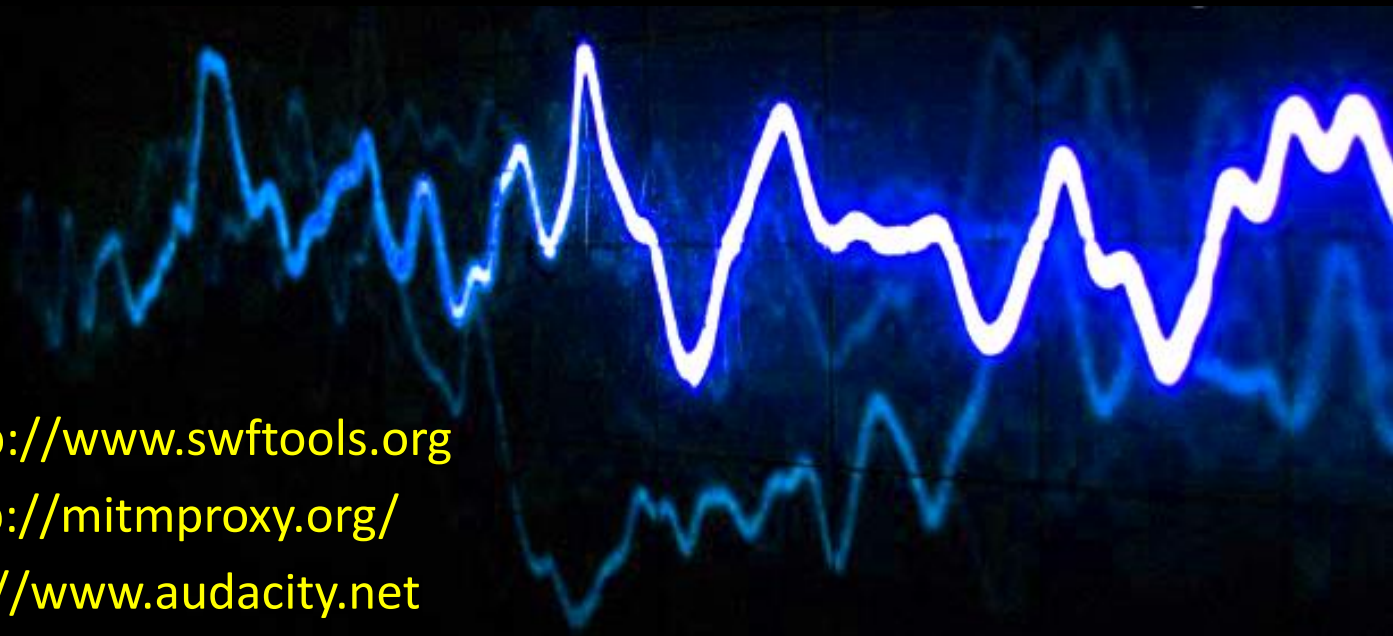
BlueVoxing
Mass Attack

1. Brute force users' mobile numbers and PINs on website
2. Wardial the found mobile numbers for weak voicemail PIN or
   - Zimbabwe telco dial +2 63 77122, enter cell number and 3 digit PIN
   - South African telco web based voice mail management system
3. Record AOTPs as voicemail greeting
4. Login to AOTP supported websites with all mobile numbers
5. Server authenticates against all the voicemail AOTPs

Thoughts

- Telephony systems do not guarantee that audio is coming from a specific handset device – implications for mobile payment systems
- Any one time password system should generate cryptographically random numbers
- Websites supporting such systems should be secure
  - OWASP top ten, SSL, brute force

References



- swfextract http://www.swftools.org
- mitmproxy http://mitmproxy.org/
- audacity http:///www.audacity.net
- ruby-processing https://github.com/jashkenas/ruby-processing/wiki
- sonic-annotater http://www.omras2.org/SonicAnnotator
- vamp audio plugins http://vamp-plugins.org/
- imacro https://addons.mozilla.org/en-us/firefox/addon/imacros-for-firefox/
- linux real time audio https://http://wiki.linuxaudio.org/wiki/real_time_inf

Thanks
Troopers13

- All Troopers
- Your attention