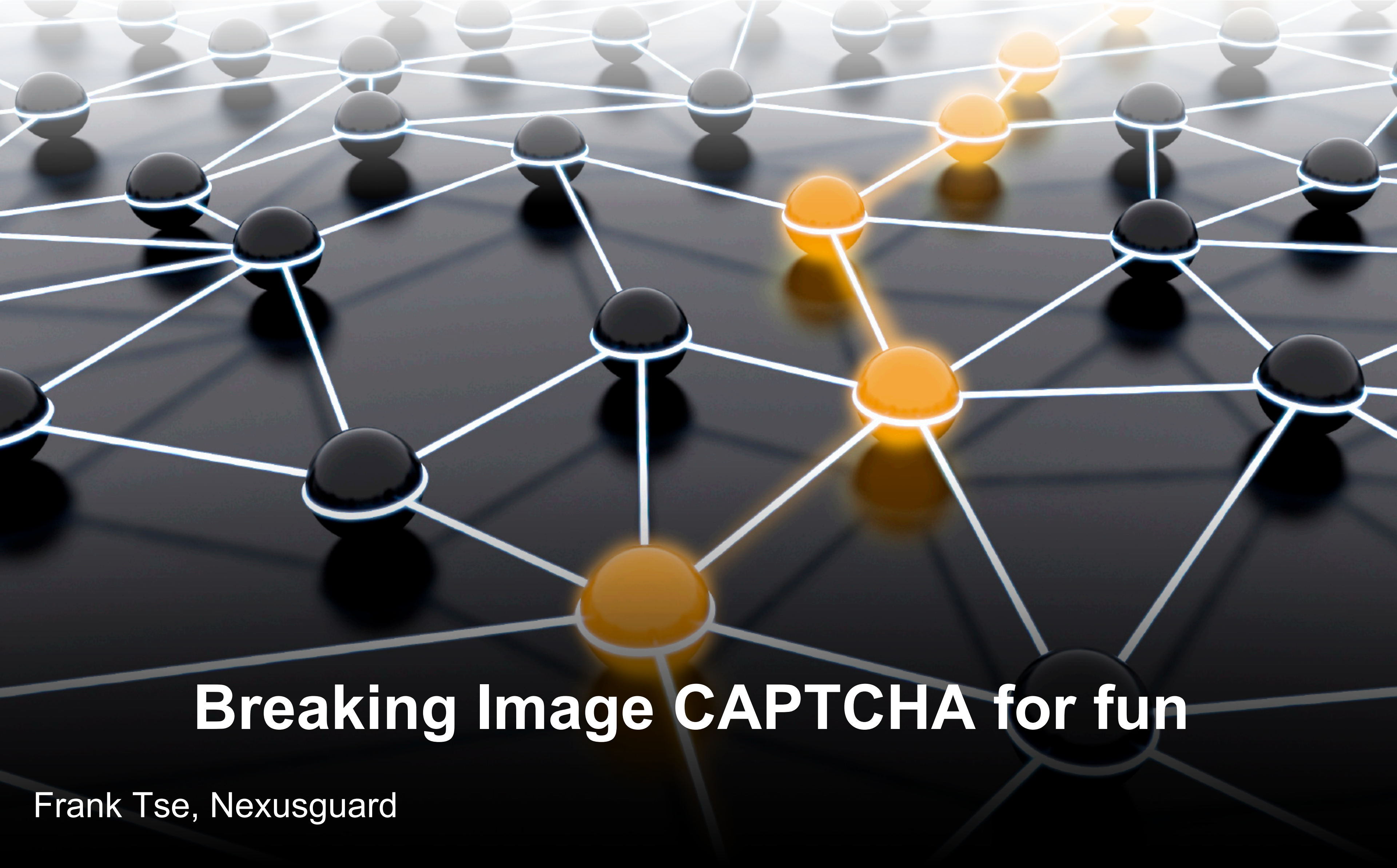
NEXUSGUARD



Breaking Image CAPTCHA for fun

Frank Tse, Nexusguard

Agenda

1 CAPTCHA and web services

2 General CAPTCHA breaking method

3 Alternative form

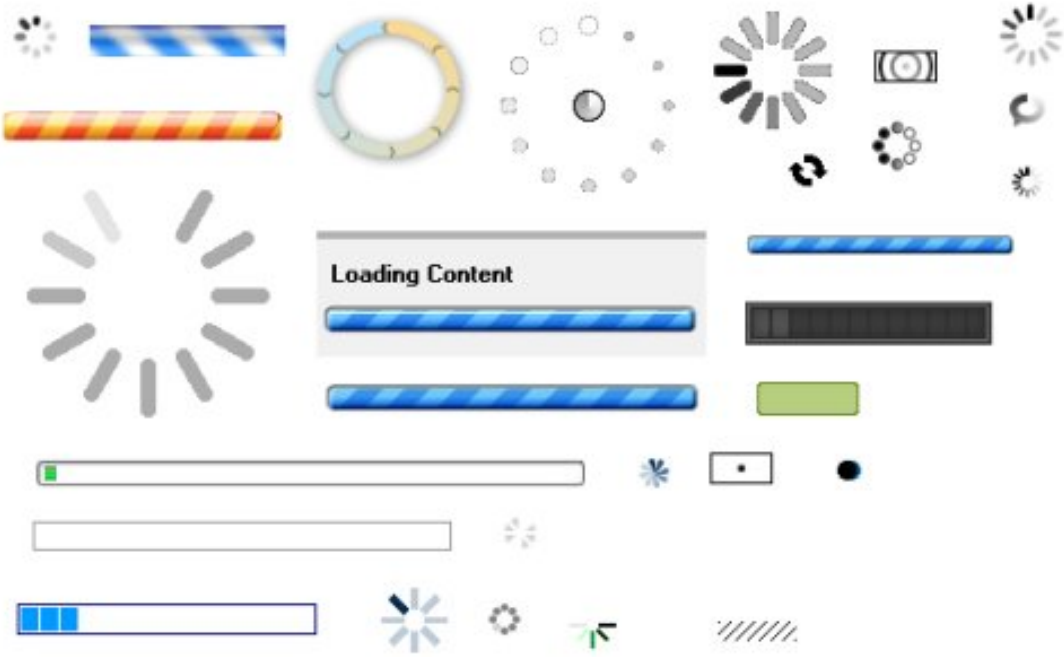
4 Analytic and optimized method

About us

- We handle DDoS attack everyday
- We face and fight with bots everyday
- Research in cryptography, imaging and coding
- Research both attack and defence methods

CAPTCHA and web services

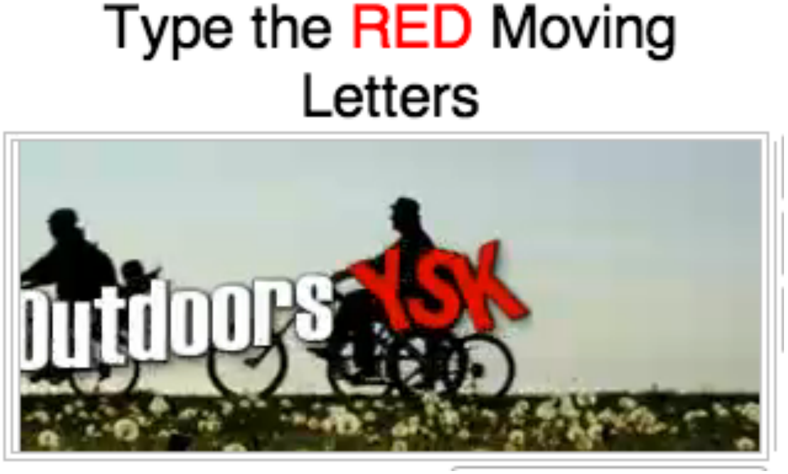
- Puzzle for machine

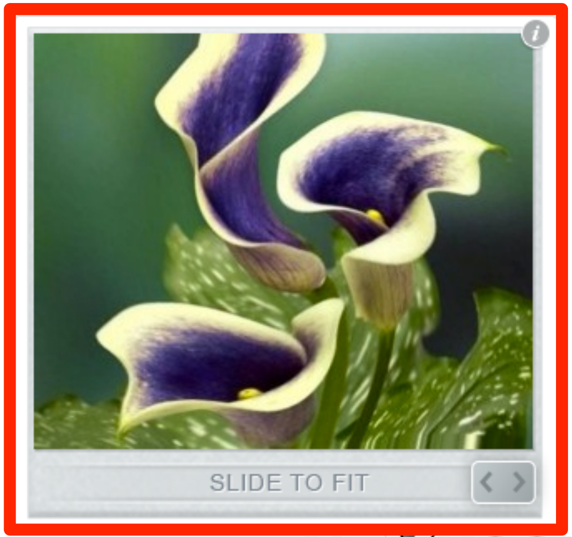


CAPTCHA and web services

- Puzzle for human

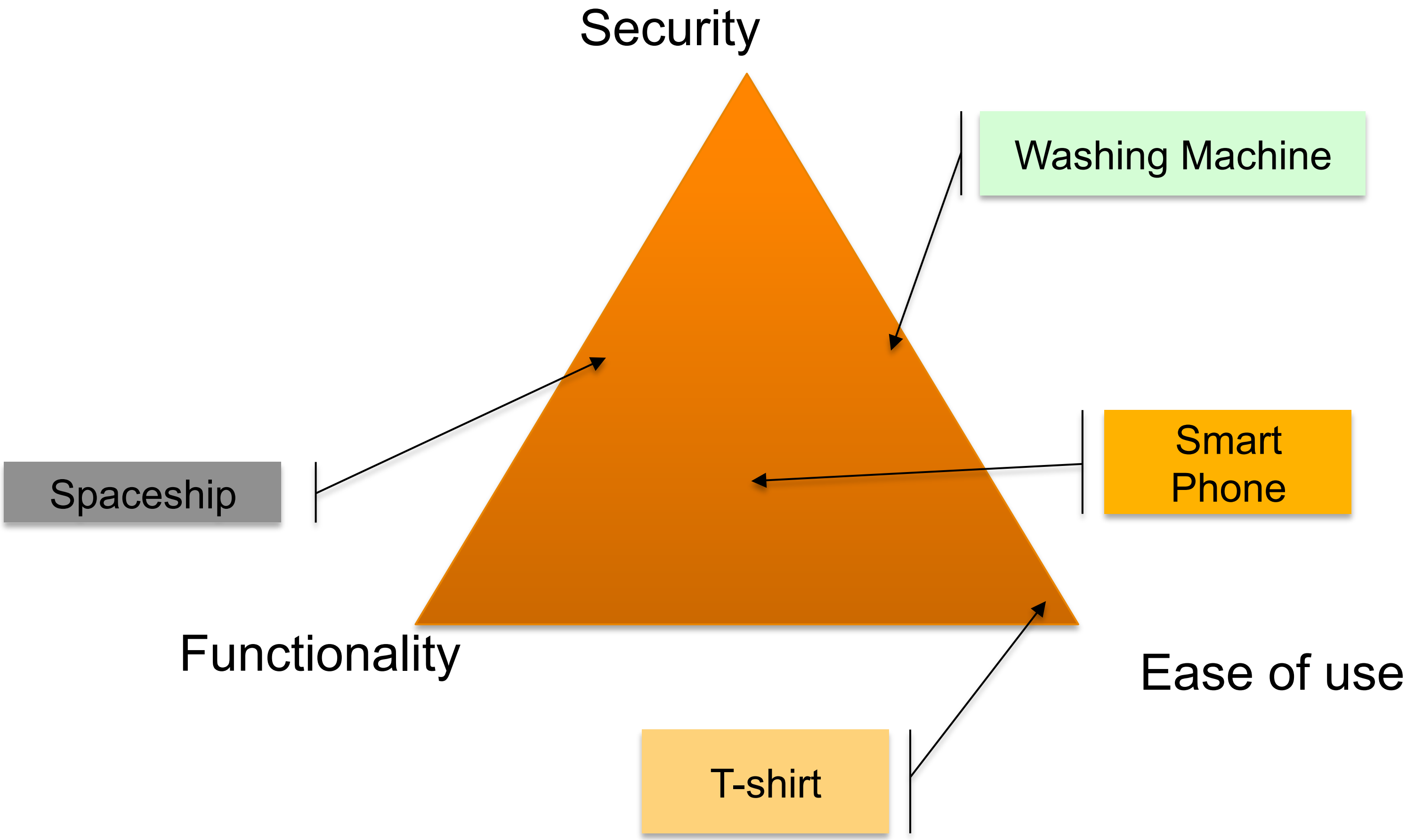
27 плюс 14 равно



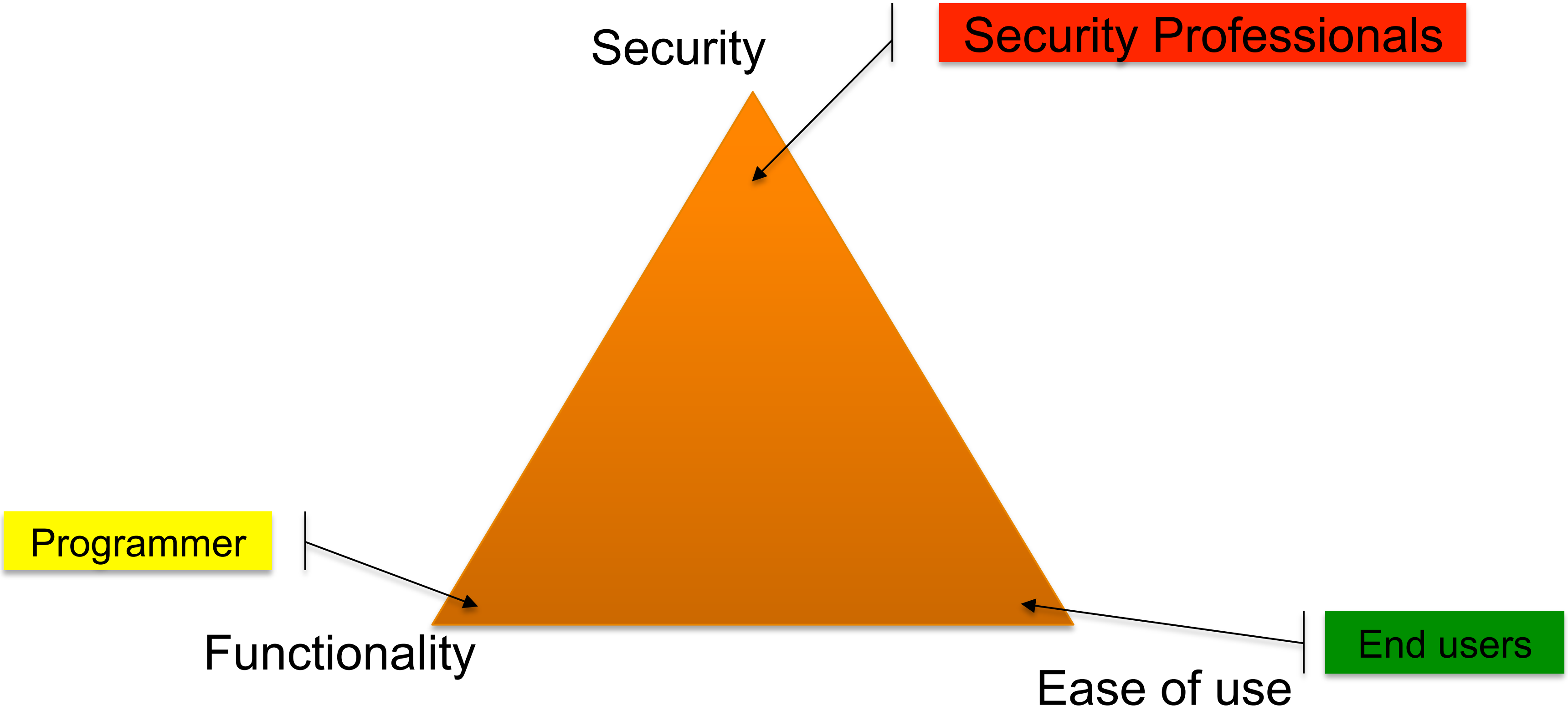


Our ~~target~~ “super star” today →

Security king ?



CAPTCHA in our eyes



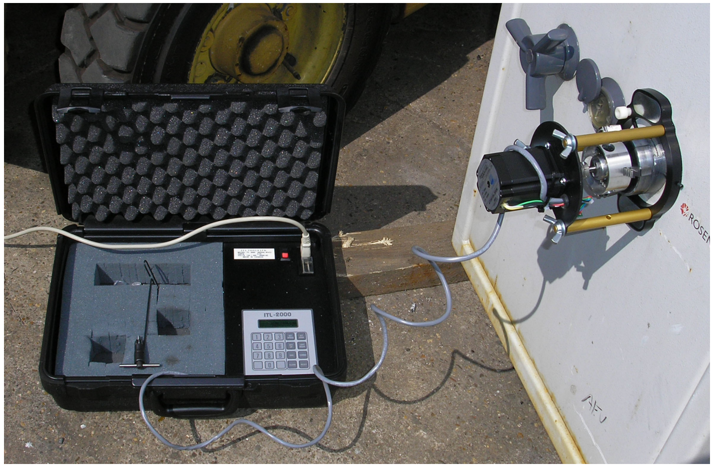
Slide-to-fit Captcha

- **The good**
  - Similar to ‘slide-to-unlock’ type authentication
  - It’s user-friendly and with higher successful rate
  - Works fine with HTML5 without Flash
  - I pick it because it responses to attackers
  - Opportunity for advertisers and sponsors
- **The bad**
  - Heavy traffic loading (~30 Images)
  - Easy to break by nature
  - Single tier, single image transformation type

General CAPTCHA breaking method

- **Lock breaking**
  - Bypass
  - Skill
  - Brute force

http://paxtonlocksmithing.com/blog/2012/02/20/credit-cards-used-to-open-doors/

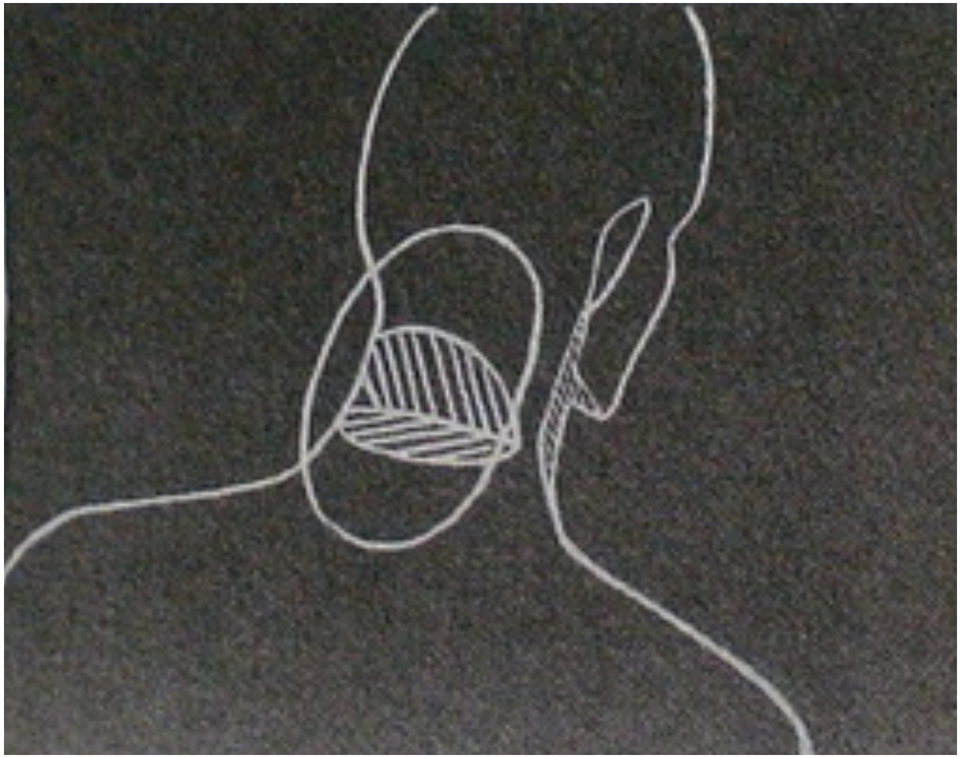


http://toool.nl/blackbag/images/itl2000.jpg

http://seattlelocksmith.net/blog/5-top-lockpicking-tools/

General CAPTCHA breaking method

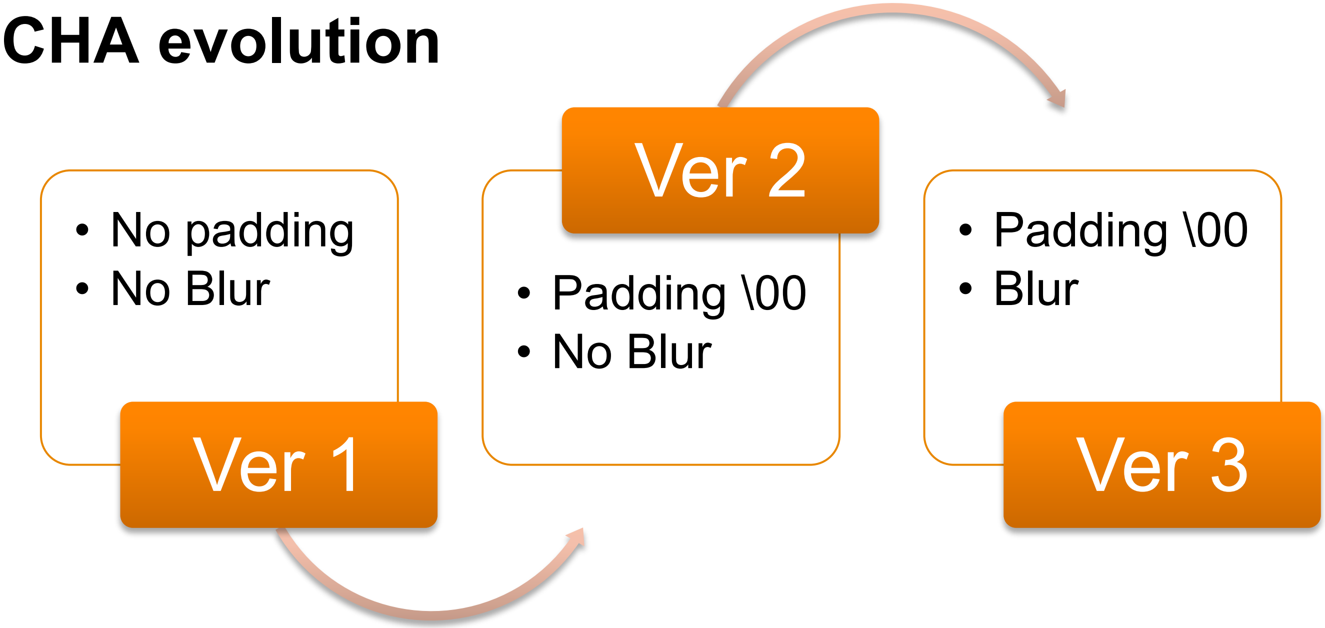
- **CAPTCHA breaking**
  - Bypass
    - Alternative form
  - Skill
    - OCR
    - Statistic
    - Curve-fitting (FFT)
    - Analytic
  - Brute force
    - Database matching
    - Effective brute force



Some academic stuffs

- **Fast Fourier Transform (FFT)**
  - Calculate how ‘blur’ the image is
- **Histogram**
  - Distribution of data by frequency (photo lighting)
  - Used to detect artificial background
- **Longest path-finding**
  - Opposite to ‘shortest path’ by Dijkstra’s Algorithm
  - Used to detect how serious the image was twisted

Image CAPTCHA evolution



Ver 1

- No padding
- No Blur

Ver 2

- Padding \00
- No Blur

Ver 3

- Padding \00
- Blur

Attack Method	Effectiveness		
Alternative Form	Good	Good	Good
Simple Statistic	Great	Poor	Poor
Modified statistic	Great	Great	Poor
FFT	Great	Poor	Poor
Analytic (Path, BG)	Great	Great	Great

Alternative form

- According to W3C Web Content Accessibility Guide (WCAG 2.0) aka ISO/IEC 40500:2012
  - Guideline 1.1 Text Alternatives
    - 1.1.1 Non-text Content: All non-text content that is presented to the user has a text alternative that serves the equivalent purpose, except for the situations listed below. (Level A)
      - CAPTCHA: If the purpose of non-text content is to confirm that content is being accessed by a person rather than a computer, then text alternatives that identify and describe the purpose of the non-text content are provided, and alternative forms of CAPTCHA using output modes for different types of sensory perception are provided to accommodate different disabilities.

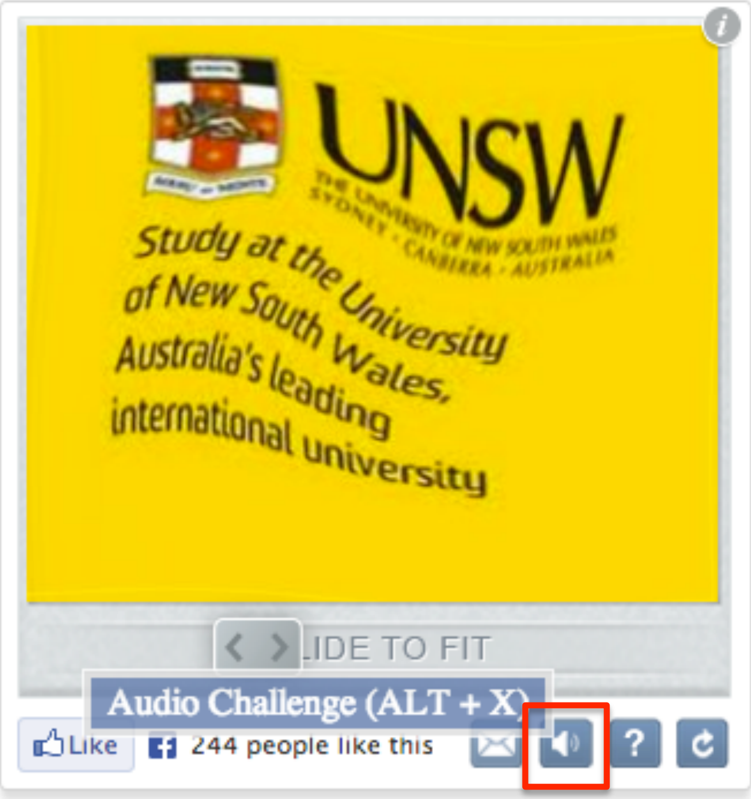
- Attack on the **weakest** alternative form

Alternative form

- **Google Voice API**
  - Pre-recorded female voice

- **Indicates the direction of correct image**
  - Slide right / left
  - Slide slightly right / left
  - You are on the right image

  - Voice is very user-friendly
  - Voice can be recognized by Google Speech-to-text and convert to text ☺

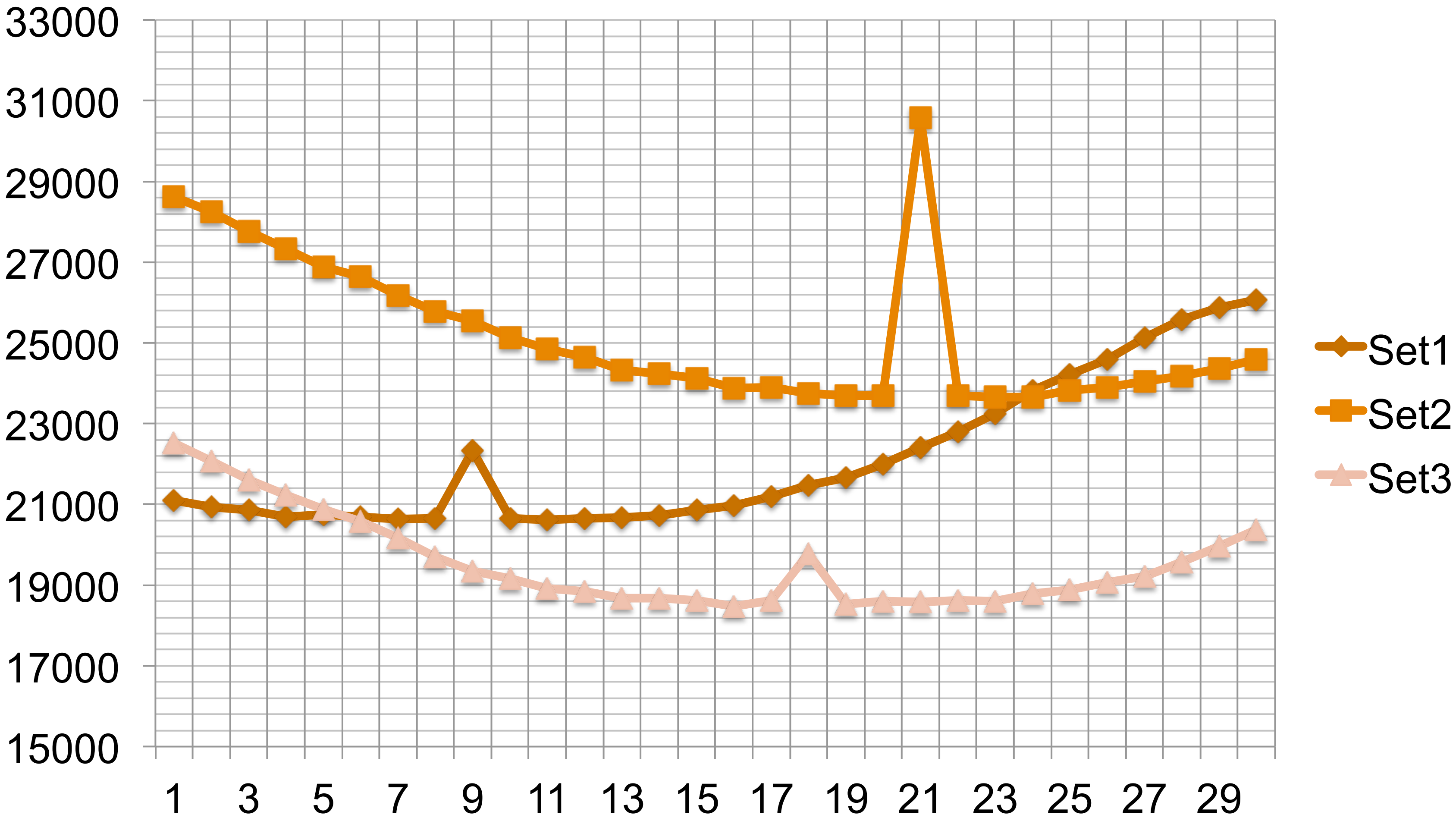

UNSW
THE UNIVERSITY OF NEW SOUTH WALES
SYDNEY · CANBERRA · AUSTRALIA

Study at the University of New South Wales, Australia's leading international university

< > LIDE TO FIT
Audio Challenge (ALT + X)
Like
244 people like this
?

NEXUSGUARD

Image File Size

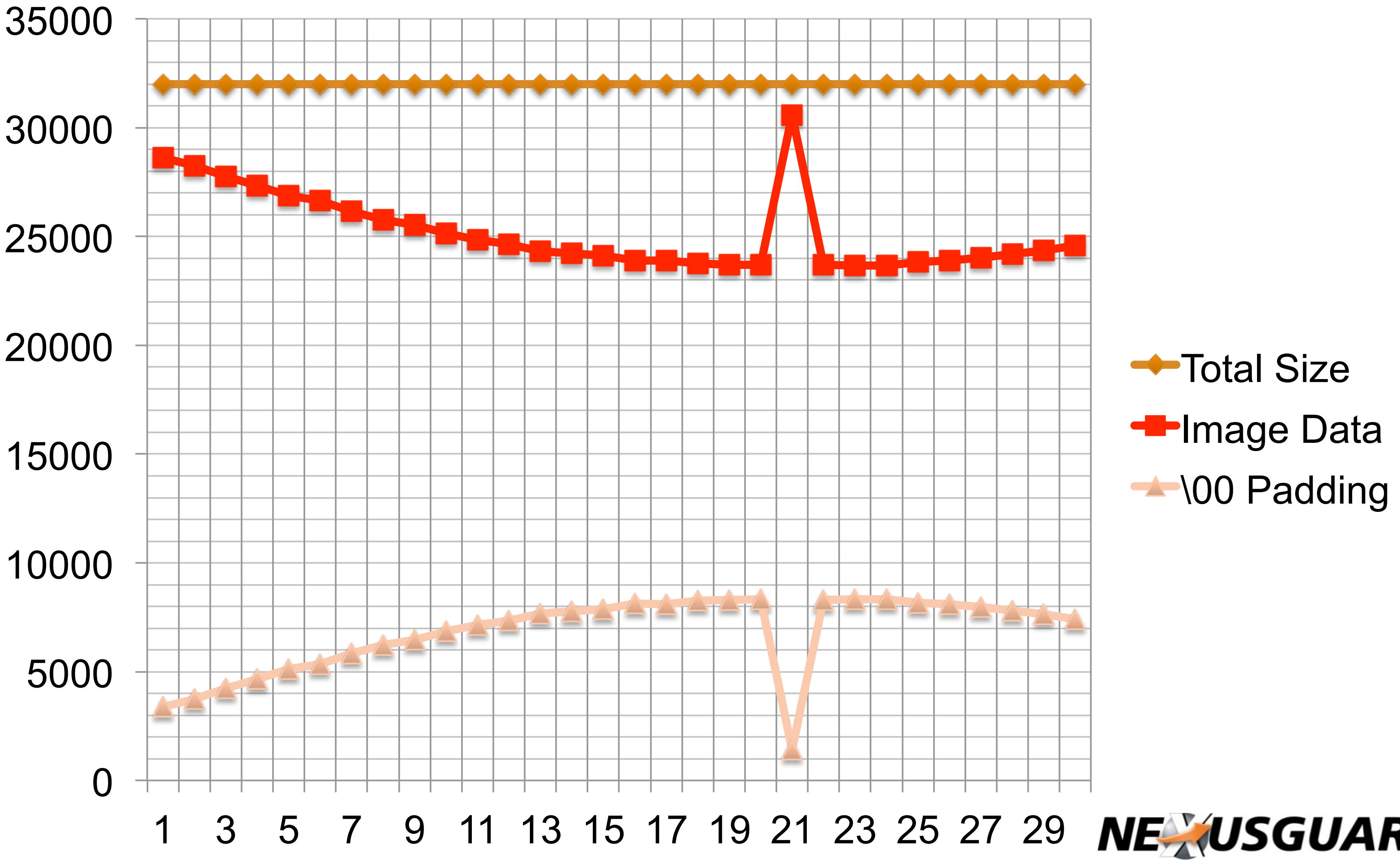


NEXUSGUARD

Optimizing the algorithm

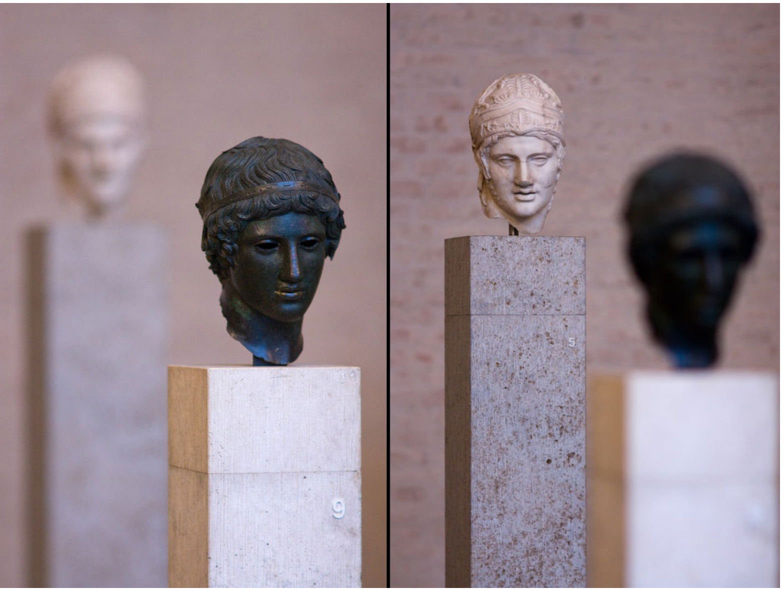
- **The Key-space**
  - Traditional CAPTCHA: 1 out of ~36^n
    - (0.00006 % for brute force when n=4)
  - Slide-to-fit : 5 out 31
    - 16% by blind brute-force
    - Correct image at border (1-3 or 28-31) is about 7%
- **Use HTTP HEAD instead of GET**
  - Image size was included in header
  - Bandwidth saved for 99%
- **Get only partial of the whole image set**
  - Getting min of 5 sample images, 95% of answers are correct
  - All linear transformation can be solved by shortcut

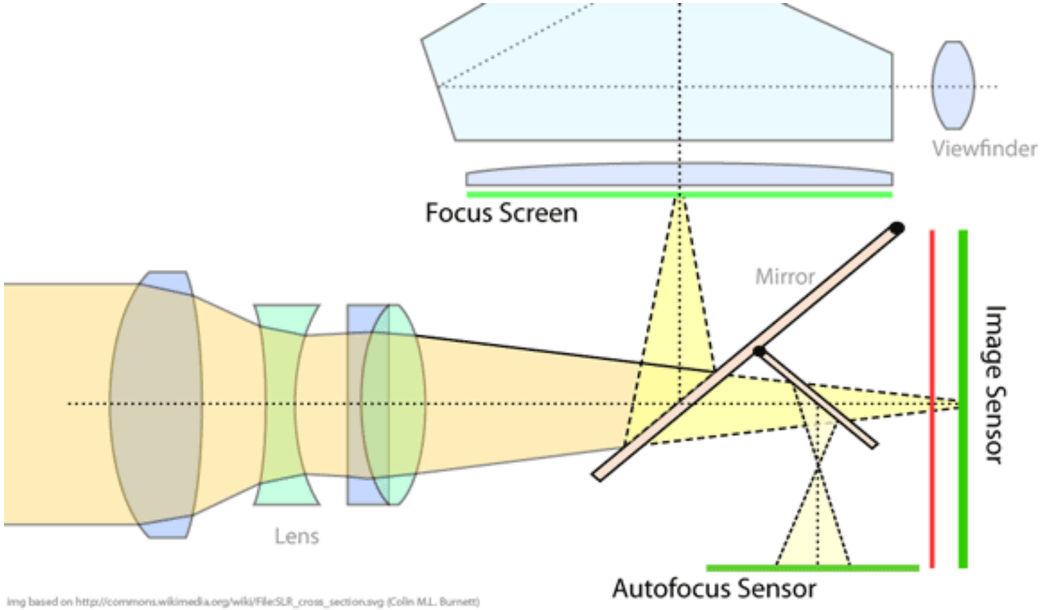
Image File Size with \00 Padding



NEXUSGUARD

Contrast Detection



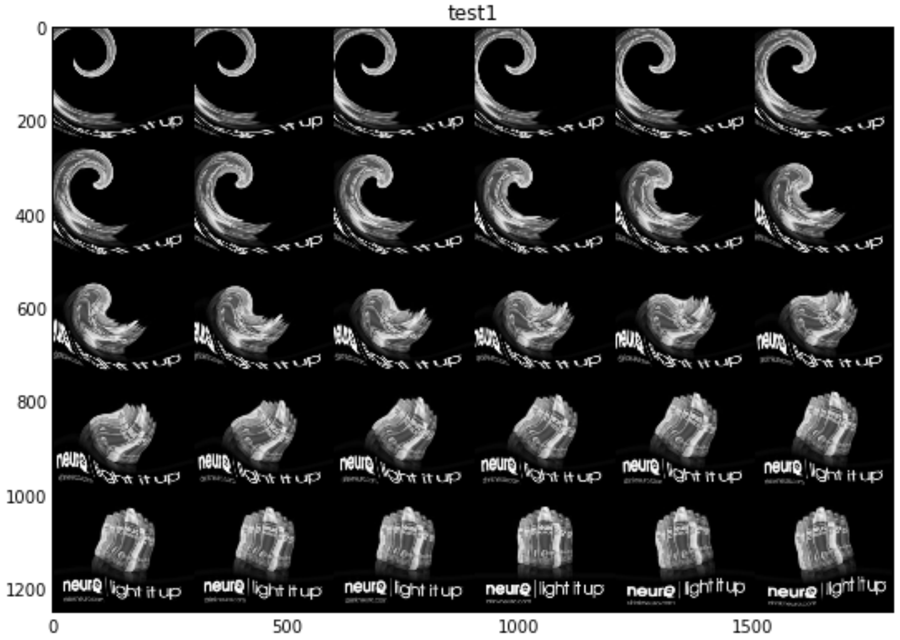


img based on http://commons.wikimedia.org/wiki/File:SLR_cross_section.svg (Colin M.L. Burnett)

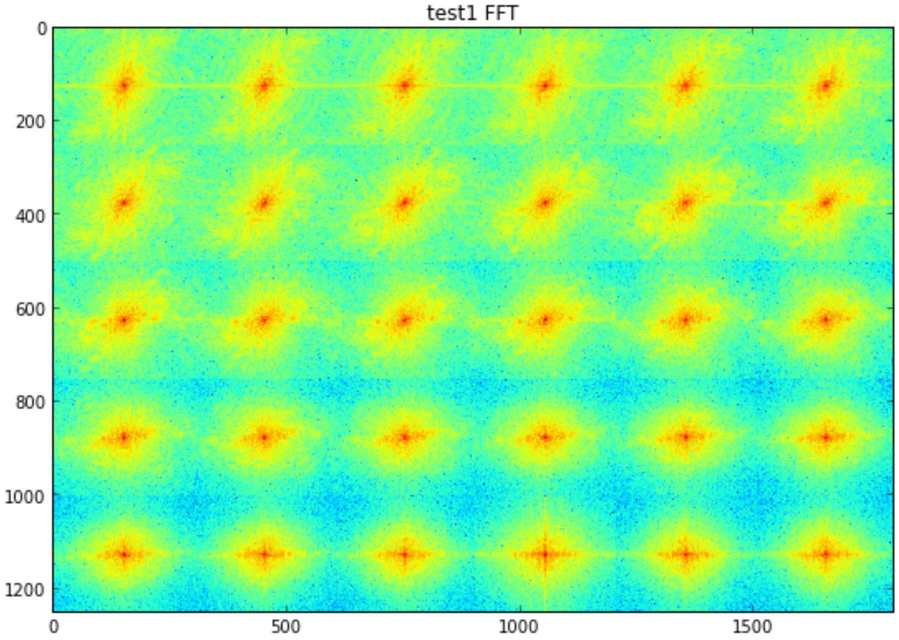
Contrast Detection

- **Rule #1**
  - Contrast of an image will reduce when it's processed with lossy-compression
- **Rule #2**
  - Contrast is calculated by differences of adjacent image points
- **Rule #3**
  - Contrast didn't care about color
- **Rule #4**
  - Image with higher sum of contrast is usually sharp

Contrast Detection



Inspected images



Contrast

Well, we make the correct image

"not that contrast"

by lossy JPEG compression

NEXUSGUARD

Image File Size with \00 Padding & not that contrast

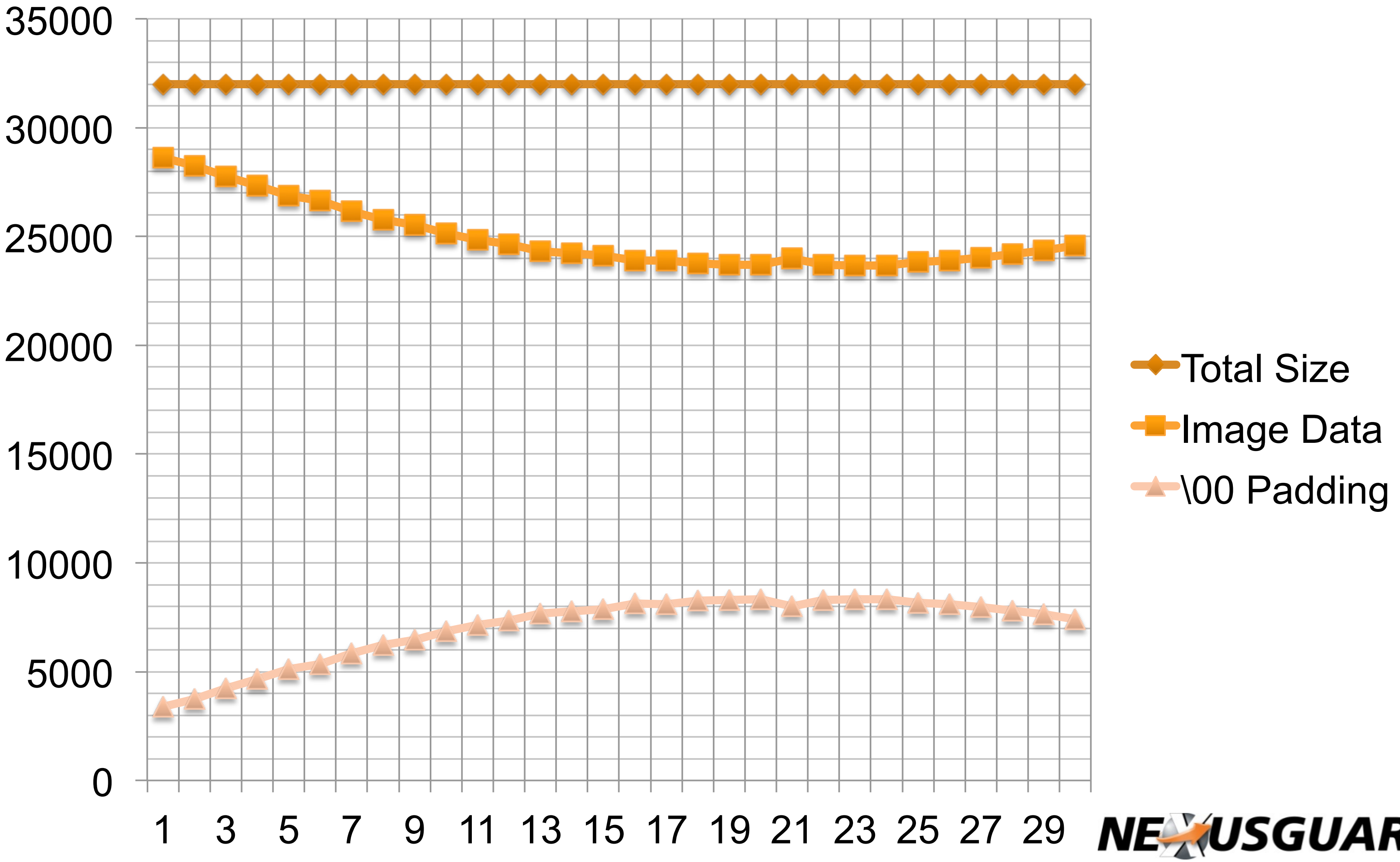
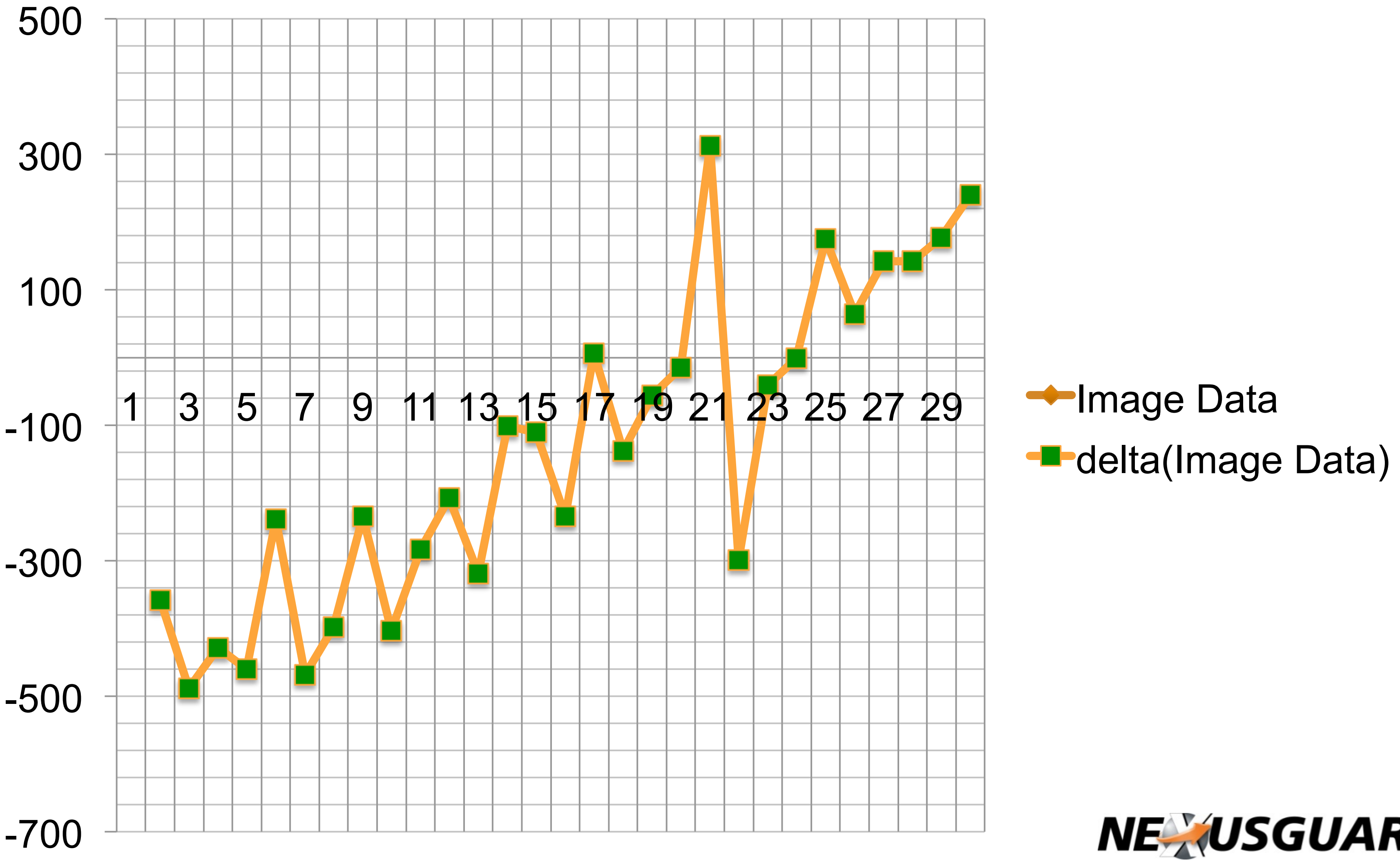


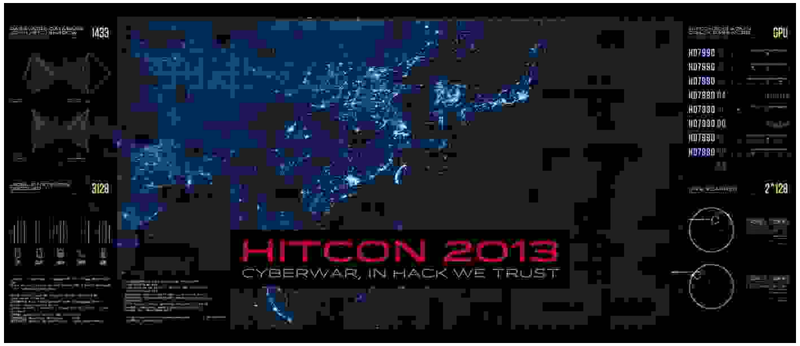
Image File Size with \00 Padding & not that contrast



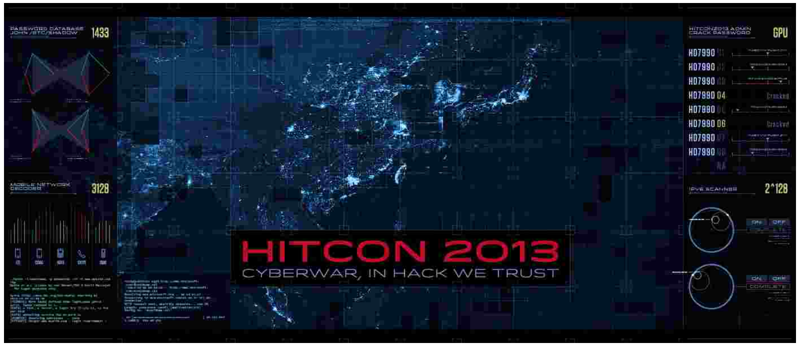
Well, we make the ALL images
in similar size
by lossy JPEG compression
with target size

```
// Generate JPG file with targeted file size
// jpg_size.py
target_size = sys.argv[1]
jpg_ql = 0
jpg_qh = 100

ε = 200 // bytes
steps = 10

while (steps >0):
        current_quality = (jpg_ql+jpg_qh)/2
        current_size = sizeof(jpg_compress(img, current_quality))
        if ( abs(current_size – target_size) < ε ): break
        if ( current_size > target_size ): jpg_qh = current_quality
        if ( current_size < target_size ): jpg_lh = current_quality
        steps-=1

output = jpg_compress(img,current_quality)
```

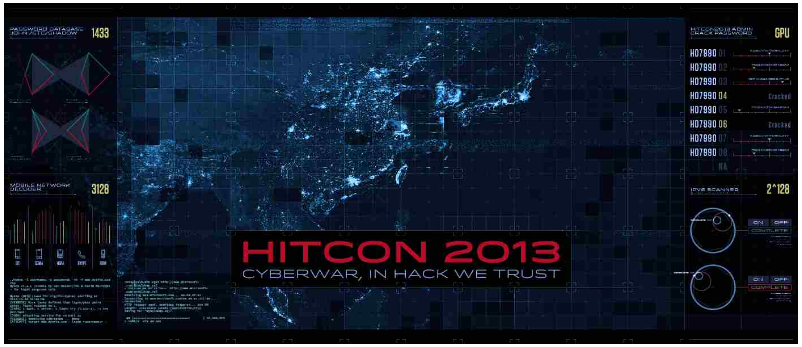
```
// Generate JPG file with ranged random target
file size
μ = 80000 // mean of target size
σ = 400 // standard deviation of target size

target_size[i] = μ + σ*(random.random())
```



18.868K/20K



79.479K/80K

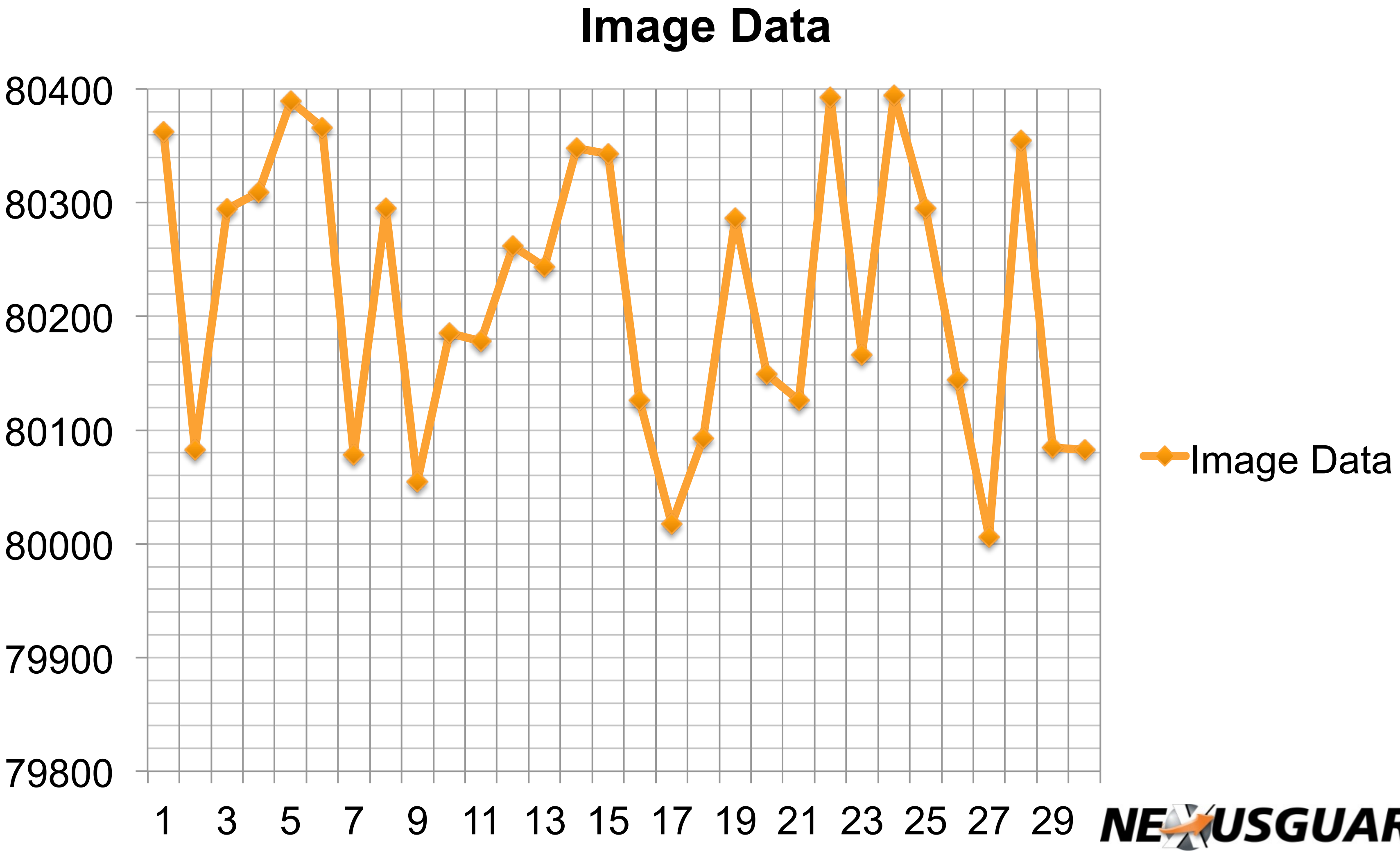


68.856K/70K

HITCON 2013
CYBERWAR, IN HACK WE TRUST

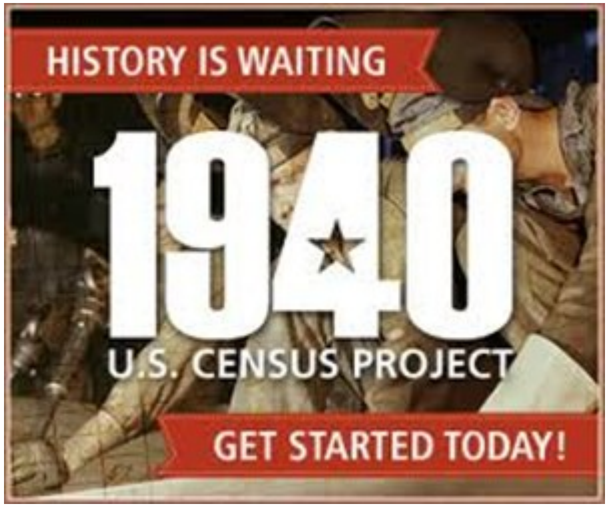
Org: 679.54K

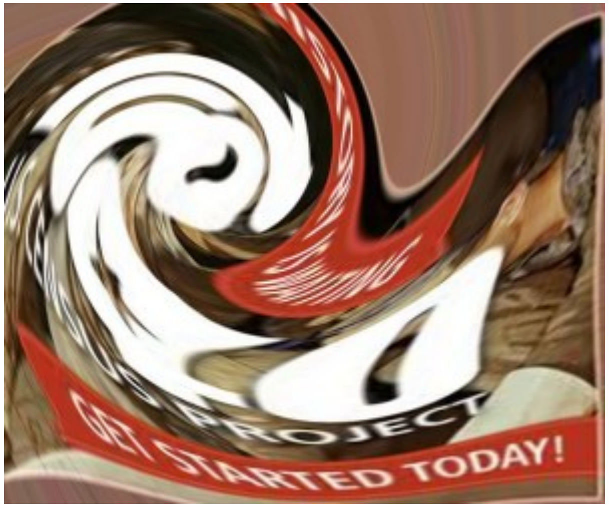
Target size 80K w/ sd 400



Analytic

- **Solution #1**
  - The background
    - Background need to be filled when twisted
    - Complementary color or patterns can be detected
- **Solution #2**
  - The boundary
    - Twisted image got longer boundary
- **Solution #3**
  - The differences
    - Side images are tended to converge to original image,
    - $\Sigma (|\Delta(img[i] - img[i+1])|)$ converges to minimum near correct image
    - Compare data uses all colour data





Do You Have Any Questions?

Contact us at: contact@nexusguard.com

NEXUSGUARD