Quantifying the computational security of multi-user systems
(Work with M. Christiansen, F. du Pin Calmon & M. Médard)

Ken Duffy

Hamilton Institute,
National University of Ireland Maynooth

Eurandom, July 2014
A stochastic network (?)

Quantifying the computational security of multi-user systems

M. Christiansen
F. du Pin Calmon (MIT)
M. Médard (MIT)

A model of computational security

*Computationally secure:*

- User selects $X$, a string, from a collection of possibilities.
- Inquisitor knows the collection of all objects and can query each in turn.
- *Computationally secure* if collection of keys is large.
A model of computational security

**Computationally secure:**
- User selects $X$, a string, from a collection of possibilities.
- Inquisitor knows the collection of all objects and can query each in turn.
- Computationally secure if collection of keys is large.

**Probability:**
- What if $X$ is picked probabilistically with a distribution known to the inquisitor?
Why non uniform?

Investigating the Distribution of Password Choices

David Malone
Hamilton Institute, National University of Ireland Maynooth
David.Malone@nuim.ie

Kevin Maher
Hamilton Institute, National University of Ireland Maynooth
Kevin.J.Maher@nuim.ie

### Why non uniform?

<table>
<thead>
<tr>
<th>Rank</th>
<th>Cyphertext</th>
<th>Indicative Hint</th>
<th>Inferred Password</th>
<th>#Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EQ7fIpT7i/Q=</td>
<td>One to six in numeral form</td>
<td>123456</td>
<td>1905308</td>
</tr>
<tr>
<td>2</td>
<td>j9p+HwtWWT86aMjgZFLzYg==</td>
<td>1234567890 ohne 0</td>
<td>123456789</td>
<td>445971</td>
</tr>
<tr>
<td>3</td>
<td>L8qbAD3j13jioxG6CatHBw==</td>
<td>Answer is password</td>
<td>password</td>
<td>343956</td>
</tr>
<tr>
<td>4</td>
<td>BB4e6X+b2xLioxG6CatHBw==</td>
<td>adbeandonetwothree</td>
<td>adobe123</td>
<td>210932</td>
</tr>
<tr>
<td>5</td>
<td>j9p+HwtWWT/ioxG6CatHBw==</td>
<td>123456789 minus last number</td>
<td>12345678</td>
<td>201150</td>
</tr>
<tr>
<td>6</td>
<td>5djv7ZCI2ws=</td>
<td>1st 123456 letters</td>
<td>qwerty</td>
<td>130401</td>
</tr>
<tr>
<td>7</td>
<td>dQ10asWPYvQ=</td>
<td>1234567 is the password</td>
<td>1234567</td>
<td>124177</td>
</tr>
<tr>
<td>8</td>
<td>7LqYzKVeQ8I=</td>
<td>6 number 1s</td>
<td>111111</td>
<td>113684</td>
</tr>
<tr>
<td>9</td>
<td>PMDtbP0LZxu03SwrFUvYGA==</td>
<td>adobe photo editing software</td>
<td>photoshop</td>
<td>83269</td>
</tr>
<tr>
<td>10</td>
<td>e6MPXQ5G6a8=</td>
<td>one two three one two three</td>
<td>123123</td>
<td>82606</td>
</tr>
</tbody>
</table>

Table III: Top 10 Adobe passwords.

---

Why non uniform?

Table III: Top 10 Adobe passwords.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Cyphertext</th>
<th>Indicative hint</th>
<th>Inferred password</th>
<th>#users</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EQ7fIpT7i/Q=</td>
<td>0ne to six in numeral form</td>
<td>123456</td>
<td>1905308</td>
</tr>
<tr>
<td>2</td>
<td>j9p+HwtWWT56aMjgZFLzYg==</td>
<td>1234567890 ohne 0</td>
<td>123456789</td>
<td>445971</td>
</tr>
<tr>
<td>3</td>
<td>LqqbAD3j13jioxG6CatHBw==</td>
<td>answer is password</td>
<td>123456788</td>
<td>343956</td>
</tr>
<tr>
<td>4</td>
<td>BB4e6X+b2xLioxG6CatHBw==</td>
<td>adbeandonetwothree</td>
<td>12345678</td>
<td>210932</td>
</tr>
<tr>
<td>5</td>
<td>j9p+HwtWWT/ioxG6CatHBw==</td>
<td>123456789 minus last number</td>
<td>123456780</td>
<td>201150</td>
</tr>
<tr>
<td>6</td>
<td>5d4v72CI2ws=</td>
<td>1st 123456 letters</td>
<td>123456787</td>
<td>130401</td>
</tr>
<tr>
<td>7</td>
<td>dQ10asWPyvG=</td>
<td>123456 is the password</td>
<td>12345674</td>
<td>124177</td>
</tr>
<tr>
<td>8</td>
<td>7LqYzXveq8I=</td>
<td>6 number 1s</td>
<td>111111</td>
<td>113684</td>
</tr>
<tr>
<td>9</td>
<td>PMDTbPOLZxu03SwFvYGA==</td>
<td>adobe photo editing software</td>
<td>110111</td>
<td>83269</td>
</tr>
<tr>
<td>10</td>
<td>e6MPXQ5G6a8=</td>
<td>one two three one two three</td>
<td>123123</td>
<td>82606</td>
</tr>
</tbody>
</table>

D. Malone, tech. report, 2014
What makes a password good?

Through 20 years of effort, we’ve successfully trained everyone to use passwords that are hard for humans to remember, but easy for computers to guess.

scriptsizekcd.com/936/
What makes a password good?

Through 20 years of effort, we've successfully trained everyone to use passwords that are hard for humans to remember, but easy for computers to guess.

--

scriptsizekcd.com/936/
Is Shannon Entropy the right measure?

Guessing and Entropy

James L. Massey

Signal & Info. Proc. Lab., Swiss Federal Inst. Tech, CH-8092 Zurich, Switzerland

Is Shannon Entropy the right measure?

• A word, $W$, picked from $\mathbb{A} = \{1, \ldots, m\}$, has Shannon entropy

$$H = - \sum_{i \in \mathbb{A}} P(W = i) \log P(W = i).$$

• How should the inquisitor guess $W$?

---

Is Shannon Entropy the right measure?

• A word, $W$, picked from $\mathcal{A} = \{1, \ldots, m\}$, has Shannon entropy

$$H = - \sum_{i \in \mathcal{A}} P(W = i) \log P(W = i).$$

• How should the inquisitor guess $W$? Assume

$$P(W = 1) \geq P(W = 2) \geq \ldots \geq P(W = m)$$

& guess in order:

Is Shannon Entropy the right measure?

- A word, $W$, picked from $\mathcal{A} = \{1, \ldots, m\}$, has Shannon entropy

\[
H = - \sum_{i \in \mathcal{A}} P(W = i) \log P(W = i).
\]

- How should the inquisitor guess $W$? Assume

\[
P(W = 1) \geq P(W = 2) \geq \ldots \geq P(W = m)
\]

& guess in order: the $i^{th}$ most likely word on the $i^{th}$ guess, $G : \mathcal{A} \mapsto \mathbb{N}$ such that $G(i) = i$ and

\[
E(G(W)) = \sum_{i \in \mathcal{A}} i P(W = i).
\]

---

What’s the right measure of Guesswork?

An Inequality on Guessing and its Application to Sequential Decoding

Erdal Arikan, Senior Member, IEEE

What’s the right measure of Guesswork?

A sequence \( W_k \in \mathbb{A}^k \) made of i.i.d. letters. Define Rényi entropy

\[
R_1(\beta) = \frac{1}{1 - \beta} \log \sum_{w \in \mathbb{A}} P(W_1 = w)^\beta,
\]

---

What’s the right measure of Guesswork?

A sequence $W_k \in \mathbb{A}^k$ made of i.i.d. letters. Define Rényi entropy

$$R_1(\beta) = \frac{1}{1 - \beta} \log \sum_{w \in \mathbb{A}} P(W_1 = w)^\beta,$$

Arikan’s Proposition:

$$\lim_{k \to \infty} \frac{1}{k} \log \mathbb{E}(G(W_k)^\alpha) = \alpha R_1 \left( \frac{1}{1 + \alpha} \right) \text{ for } \alpha > 0.$$
What’s the right measure of Guesswork?

E.g. $\alpha = 1$, for large $k$

$\mathbb{E}(G(W_k)) \approx \exp(kR_1(1/2))$

where

$$R_1(1/2) = \log \left( \sum_{w \in A} \sqrt{P(W_1 = w)} \right)^2.$$

E.g. Bernoulli Source, log base 2.

---

Source generalization of Arikan’s Proposition

With the Rényi entropy of $W_k$ being

$$R_k(\beta) = \frac{1}{1-\beta} \log \sum_{w \in \mathbb{A}^k} P(W_k = w)^\beta,$$

and $R(\beta) = \lim_{k \to \infty} \frac{1}{k} R_k(\beta)$, generalizations prove

$$\lim_{k \to \infty} \frac{1}{k} \log \mathbb{E}(G(W_k)^\alpha) = \alpha R \left( \frac{1}{1 + \alpha} \right) \quad \text{for} \quad \alpha > -1.$$
Source generalization of Arikan’s Proposition

With the Rényi entropy of $W_k$ being

$$R_k(\beta) = \frac{1}{1 - \beta} \log \sum_{w \in A^k} P(W_k = w)^{\beta},$$

and $R(\beta) = \lim_{k \to \infty} \frac{1}{k} R_k(\beta)$, generalizations prove

$$\lim_{k \to \infty} \frac{1}{k} \log \mathbb{E}(G(W_k)^\alpha) = \begin{cases} \alpha R \left( \frac{1}{1 + \alpha} \right) & \text{for } \alpha > -1, \\ -R(\infty) & \text{for } \alpha \leq -1. \end{cases}$$

---

Large deviations and guesswork distributions

Consider

\[ \Lambda(\alpha) := \lim_{k \to \infty} \frac{1}{k} \log \mathbb{E}(G(W_k)^\alpha) = \lim_{k \to \infty} \frac{1}{k} \log \mathbb{E}(e^{\alpha \log(G(W_k))}) = \begin{cases} \alpha R \left( \frac{1}{1 + \alpha} \right) \\ -R(\infty) \end{cases} \]
Large deviations and guesswork distributions

Consider

\[ \Lambda(\alpha) := \lim_{k \to \infty} \frac{1}{k} \log \mathbb{E}(G(W_k)^\alpha) = \lim_{k \to \infty} \frac{1}{k} \log \mathbb{E}(e^{\alpha \log(G(W_k))}) = \begin{cases} \alpha R \left( \frac{1}{1 + \alpha} \right) \\ -R(\infty) \end{cases} \]

Suggestive of

\[ dP \left( \frac{1}{k} \log G(W_k) \approx x \right) \propto \exp(-k\Lambda^*(x)) \, dx \]

where \( \Lambda^*(x) = \sup_{\alpha \in \mathbb{R}} (\alpha x - \Lambda_X(\alpha)) \).

For large \( k \), some jiggery-pokery gives

\[ P(G(W_k) = n) \approx \frac{1}{n} \exp \left( -k \Lambda^* \left( \frac{1}{k} \log n \right) \right). \]

---

What's in a discontinuous derivative?

\[ \Lambda(\alpha) = \begin{cases} \alpha R((1 + \alpha)^{-1}) & \text{if } \alpha \geq -1 \\ -R(\infty) & \text{if } \alpha \leq -1 \end{cases} \]

Define:

\[ \gamma = \lim_{\alpha \downarrow -1} \frac{d}{d\alpha} \Lambda(\alpha) \]

\[ = \lim_{\beta \to \infty} \left( R(\beta) - \frac{R'(\beta)}{\beta^2} \right). \]

If i.i.d., then \( \gamma = \log |\{w : P(W_1 = w) = P(G(W_1) = 1)\}|. \)
What’s in a discontinuous derivative?

\[ \Lambda(\alpha) = \begin{cases} \alpha R((1 + \alpha)^{-1}) & \text{if } \alpha \geq -1 \\ -R(\infty) & \text{if } \alpha \leq -1 \end{cases} \]

Define:

\[ \gamma = \lim_{\alpha \downarrow -1} \frac{d}{d\alpha} \Lambda(\alpha) = \lim_{\beta \to \infty} \left( R(\beta) - \frac{R'(\beta)}{\beta^2} \right). \]

If i.i.d., then \( \gamma = \log |\{w : P(W_1 = w) = P(G(W_1) = 1)\}|. \)
If not, then approximately \( e^{k\gamma} \) “most likely words” of length \( k. \)
Lemma: For $\{W_k\}$ constructed of Markovian letters with $A = \{0, 1\}$,

$$\gamma = \lim_{\alpha \downarrow -1} \Lambda'(\alpha) \in \{0, \log(\phi), \log(2)\},$$

where $\phi = (1 + \sqrt{5})/2$ is the Golden Ratio, and no other values are possible.
Uniformity, typical set coding etc.

Guessing a password over a wireless channel (on the effect of noise non-uniformity)

Mark M. Christiansen and Ken R. Duffy
Hamilton Institute
National University of Ireland, Maynooth
Email: {mark.christiansen, ken.duffy}@nuim.ie

Flávio du Pin Calmon and Muriel Médard
Research Laboratory of Electronics
Massachusetts Institute of Technology
Email: {flavio, medard}@mit.edu

Brute force searching, the typical set and Guesswork

Mark M. Christiansen and Ken R. Duffy
Hamilton Institute
National University of Ireland, Maynooth
Email: {mark.christiansen, ken.duffy}@nuim.ie

Flávio du Pin Calmon and Muriel Médard
Research Laboratory of Electronics
Massachusetts Institute of Technology
Email: {flavio, medard}@mit.edu

Multiple users

$V \in \mathbb{N}$ users, independently picking strings

$$\tilde{W}_k = (W_k^{(1)}, \ldots, W_k^{(V)}) \in \mathbb{A}^{kV}.$$ 

Statistics of each user’s selection known to an inquisitor who can query the veracity of (user, string) pair and we wishes to identify $U \leq V$ of them.
The Shannon Cipher System with a Guessing Wiretapper

Neri Merhav, Fellow, IEEE, and Erdal Arikan, Senior Member, IEEE

The Shannon Cipher System with a Guessing Wiretapper

Neri Merhav, Fellow, IEEE, and Erdal Arikan, Senior Member, IEEE

Then, it is clear that the best guessing strategy (in any reasonable sense) is to first guess the most likely $X$ given $Y$, then try the second most likely guess, and so on, until eventually, the correct message is found.

Optimal strategy?

$G$ is optimal $W_k$ if and only if

$$P(G(W_k) \leq n) \geq P(S(W_k) \leq n)$$

for all strategies $S$ and all $n \in \{1, \ldots, m^k\}$. 
Optimal strategy?

$G$ is optimal $W_k$ if and only if

$$P(G(W_k) \leq n) \geq P(S(W_k) \leq n)$$

for all strategies $S$ and all $n \in \{1, \ldots, m^k\}$.

Lemma

If $V = U$, the optimal strategies are those that guess from most likely to least likely.
If $U < V$, not guaranteed stochastic domination

Example: $V = 2$, $U = 1$ and $|A| = 3$. 
If $U < V$, not guaranteed stochastic domination

Example: $V = 2$, $U = 1$ and $|A| = 3$.

<table>
<thead>
<tr>
<th>User</th>
<th>Item 1</th>
<th>Item 2</th>
<th>Item 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sem</td>
<td>0.6</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>Johan</td>
<td>0.6</td>
<td>0.2</td>
<td>0.2</td>
</tr>
</tbody>
</table>
If $U < V$, not guaranteed stochastic domination

Example: $V = 2$, $U = 1$ and $|A| = 3$.

<table>
<thead>
<tr>
<th>User</th>
<th>Item 1</th>
<th>Item 2</th>
<th>Item 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sem</td>
<td>0.6</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Johan</td>
<td>0.6</td>
<td>0.2</td>
<td>0.2</td>
</tr>
</tbody>
</table>
If $U < V$, not guaranteed stochastic domination

Example: $V = 2$, $U = 1$ and $|A| = 3$.

<table>
<thead>
<tr>
<th>User</th>
<th>Item</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sem</td>
<td>0.6</td>
<td>0.5</td>
</tr>
<tr>
<td>Johan</td>
<td>0.6</td>
<td>0.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User</th>
<th>Item</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sem</td>
<td>0.6</td>
<td>0.5</td>
</tr>
<tr>
<td>Johan</td>
<td>0.6</td>
<td>0.2</td>
</tr>
</tbody>
</table>
There exist asymptotically optimal strategies - Round-robin

For each \( v \in \{1, \ldots, V\} \) let \( G^{(v)} \) denote its optimal strategy and define:

\[
G_{\text{opt}}(U, V, \vec{W}_k) = \text{U-min} \left( G^{(1)}(W^{(1)}_k), \ldots, G^{(V)}(W^{(V)}_k) \right),
\]

where \( \text{U-min} : \mathbb{R}^V \to \mathbb{R} \) gives the \( U^{th} \) smallest component.
There exist asymptotically optimal strategies - Round-robin

For each \( v \in \{1, \ldots, V\} \) let \( G^{(v)} \) denote its optimal strategy and define:

\[
G_{\text{opt}}(U, V, \vec{W}_k) = \text{U-min} \left( G^{(1)}(W^{(1)}_k), \ldots, G^{(V)}(W^{(V)}_k) \right),
\]

where \( \text{U-min} : \mathbb{R}^V \to \mathbb{R} \) gives the \( U^{\text{th}} \) smallest component.

Then

\[
G_{\text{opt}}(U, V, \vec{W}_k) \leq \text{real performance of round-robin} \leq V G_{\text{opt}}(U, V, \vec{W}_k)
\]

and, as \( k \to \infty \), these have the same asymptote.
Asymptotically optimal strategies satisfy a LDP

Theorem

\( \{ k^{-1} \log G_{opt}(U, V, \tilde{W}_k) \} \) satisfies a large deviation principle. Defining

\[
\delta^{(v)}(x) = \begin{cases} 
\Lambda^{(v)}_G(x) & \text{if } x \leq H^{(v)} \\
0 & \text{otherwise,}
\end{cases} \quad \text{and} \quad \gamma^{(v)}(x) = \begin{cases} 
\Lambda^{(v)}_G(x) & \text{if } x \geq H^{(v)} \\
0 & \text{otherwise,}
\end{cases}
\]

the rate function is

\[
I_{G_{opt}}(U, V, x) = \max_{v_1, \ldots, v_V} \left( \Lambda^{(v_1)}_G(x) + \sum_{i=2}^{U} \delta^{(v_i)}(x) + \sum_{i=U+1}^{V} \gamma^{(v_i)}(x) \right),
\]

which may not be convex. The sCGF is

\[
\Lambda_{G_{opt}}(U, V, \alpha) = \lim_{k \to \infty} \frac{1}{k} \log E(\exp(\alpha \log G_{opt}(U, V, \tilde{W}_k)))
\]

\[= \sup_{x \in [0, Vm]} (\alpha x - I_{G_{opt}}(U, V, x)).\]
A Merhav & Arikan example, \( U = 1, \ V = 2 \)

\( W_k^{(1)} \), Bernoulli on \( \{0, 1\} \),

\[
P(W_1^{(2)} = i) = \begin{cases} 
0.55 & \text{if } i = 0 \\
0.1 & \text{if } i \in \{1, 2\} \\
0.05 & \text{if } i \in \{3, \ldots , 7\}
\end{cases}
\]
All things being equal

Corollary

If users’ statistics are all (asymptotically) the same, then

\[ \Lambda_{G_{opt}}^*(U, V, x) = \begin{cases} 
U \Lambda_G^*(x) & \text{if } x \leq H \\
(V - U + 1) \Lambda_G^*(x) & \text{if } x \geq H
\end{cases} \]

and

\[ \Lambda_{G_{opt}} (U, V, \alpha) = \begin{cases} 
U \Lambda_G \left( \frac{\alpha}{U} \right) & \text{if } \alpha \leq 0 \\
(V - U + 1) \Lambda_G \left( \frac{\alpha}{V - U + 1} \right) & \text{if } \alpha \geq 0.
\end{cases} \]
Multi-user guesswork growth rates

\[ n = V - U, \text{ number of excess strings} \]

\[ \mathbb{E}(G_{\text{opt}}(U, V, \vec{\bar{W}}_k)) \approx \exp \left( kR \left( \frac{n + 1}{n + 2} \right) \right), \text{ where } \frac{n + 1}{n + 2} \in \left\{ \frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \ldots \right\}. \]
Concluding comments

- There’s no "truly" optimal guessing strategy.
- Performance of asymptotically optimal strategies can be analysed.
- From an attacker’s point of view, there’s a law of diminishing returns in excess number of users.
- Shannon Entropy provides a universal lower bound on the guesswork growth rate of multi-user systems.
Concluding comments

- There’s no "truly" optimal guessing strategy.
- Performance of asymptotically optimal strategies can be analysed.
- From an attacker’s point of view, there’s a law of diminishing returns in excess number of users.
- Shannon Entropy provides a universal lower bound on the guesswork growth rate of multi-user systems.
- If you had an Adobe password, change it everywhere.
Same as Facebook
Same as Facebook