Data Storage and Security Strategies of Network Identity

(YOCSEF Report)

Antiy Labs

12/30/2011

Self Introduction

Identity: Xinguang, Xiao (real name); Haike, Jiang (online name); Seak (English name);

Profession: antivirus researcher, not algorithm researcher;

Major: Automation Control, not Computer Science;

English skill: poor, but like using acronyms such as AV.

Wish: do not embarrass an antivirus researcher.

Terms in This Report

- Encryption/cipher text: Encryption is the process of transforming plain text to cipher text.
- Computation speed: As for the computation speed of hash and other encryption algorithms, there is no consensus of whether we should use the length of plain text/computation time, or the computation number in unit time.
- Zhang's Theorem/ Zhang's Hypothesis: Zhang is the CTO of Antiy. Whenever any hypothesis occurs to me, I name it as "Zhang's Theorem". In order to differentiate the ideas of us two, the idea Zhang comes up with is "Zhang's Hypothesis".

Outline

- Background
- Cipher text attack methods and current solutions
- Find suitable security products
- Extra topics

When Moore Law becomes a disaster.

BACKGROUND

Computation Speed

Host Environment:
Core 2 (T7250) 2.0 G, 2M cache, 4GB RAM, 64-byte Windows server 2008

VM Environment:
Vmware Server, Ubuntu 10.04, 512MB RAM

Compute MD5 of the 16 bytes in the virtual machine, the computation number in 2.99 seconds is 1,759,393.

Computation Resources

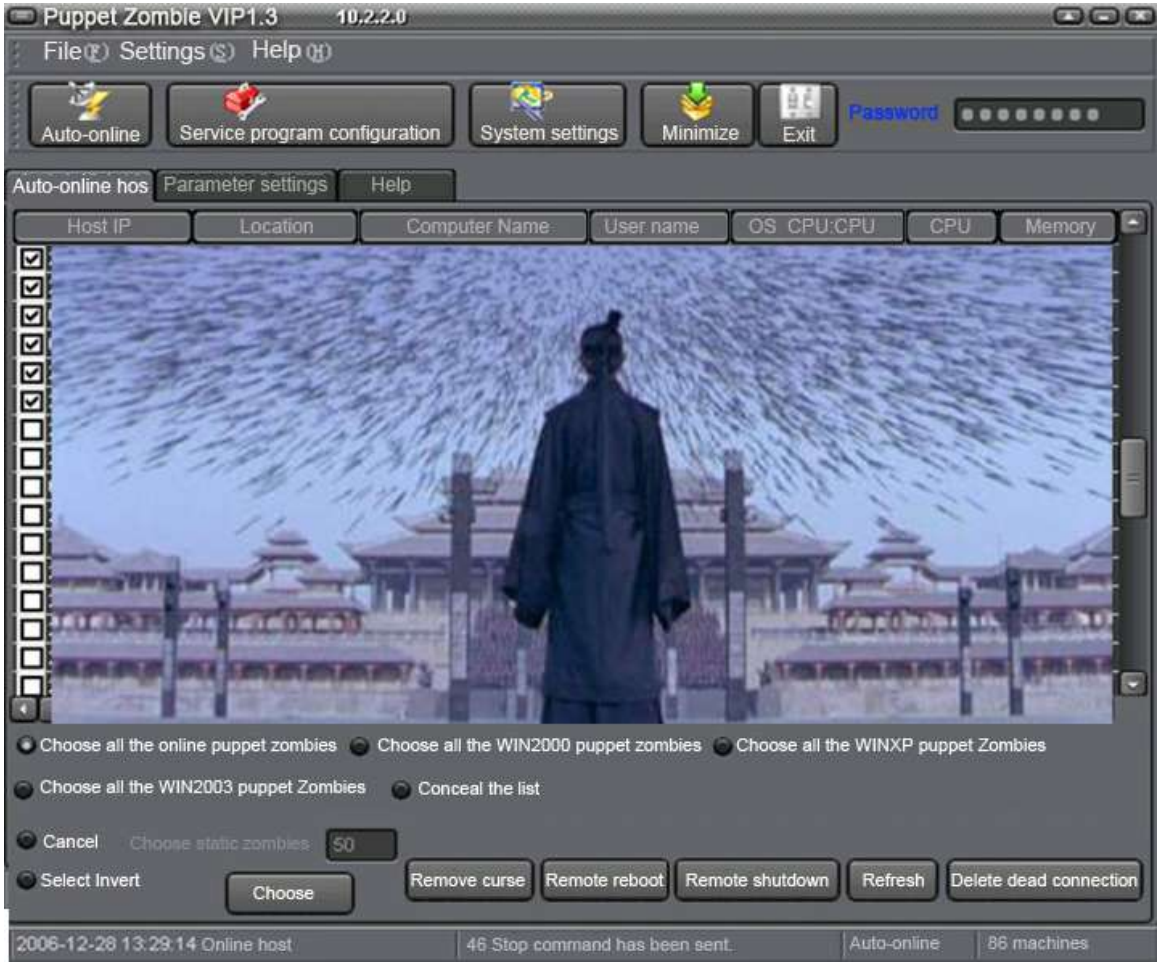
Low-cost cloud computation

amazon web services™

Super Computer: GPU

Node Resources

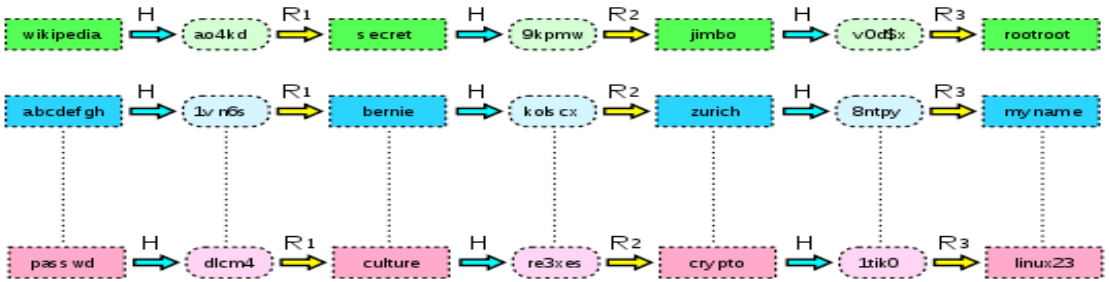
- Botnet



Plain Text Resources

Infected Vendors/Websites	Infected accounts
Sony	101,600,000
Sega Corporation	1,300,000
City Bank	200,000
CSDN	6,000,000
Duo Wan	8,000,000
Tian Ya	30,000,000

Rainbow Table/Online Query



Name	Functionality	Application scenarios
md5	Encrypt the passwords by MD5	Lots of common websites
md5(md5($pass))	Encrypt the MD5 value by MD5	Lots of common websites
md5($pass.$salt)	Connect the password with the salt to form a new password, and then encrypt the new password by MD5	open source CMS operating system such as Joomla
md5($salt.$pass)	Connect the password with the salt to form a new password, and then encrypt the new password by MD5	open source electrical business system osCommerce
md5(md5($pass).$salt)	Connect the MD5 value with the salt, and then encrypt the newly generated password by MD5	Forum systems such as Vbulletin, IceBB, and Discuz
md5(md5($salt).$pass)	Connect the MD5 value of the salt with the password, and then encrypt the newly generated password by MD5	
md5($salt.$pass.$salt)	Connect the salt with the password, and then encrypt the newly generated password by MD5	Online P2P systems such as TBDev
md5($salt.md5($pass))	Connect the salt with the MD5 value of the password, and then encrypt the newly generated password by MD5	
md5(md5($pass).md5($salt))	Connect the MD5 value of the password and the MD5 value of the salt, and then encrypt the newly generated password by MD5	
md5(md5($salt).md5($pass))	Connect the MD5 value of the salt and the MD5 value of the password, and then encrypt the newly generated password by MD5	Forum systems sucha s ipb and mybb
MD5(Unix)	Encrypt the password by MD5, and then save the value with Unix shadow format	Forum systems such as phpBB3; blog system s such as WordPress
md5(unicode)	Encrypt the Unicode of the password by MD5	Unix/Linux systems
sha1	Encrypt the password by SHA-1	Lots of common websites
sha1($salt.$pass)	Connect the salt with the password, and then encrypt the newly generated password by SHA-1	
sha1(lower($username).$pass)	Connect the lower-case user name with the password, and then encrypt the newly generated password by SHA-1	Forum systems such as SMF
sha1(upper($username).':'.upper($pass))	Connect the upper-case user name with ":", and the upper-case password, and then encrypt the newly generated password by SHA-1	Online game server systems such as ManGOS
sha1($username.':'.$pass)	Connect the user name with ":" and the password, and then encrypt the newly generated password by SHA-1	
sha256	Encrypt the password by SHA-256	
sha512	Encrypt the password by SHA-512	
mysql	Encrypt the password of MySQL accounts	MySQL database (Version 4.0 and earlier versions)
mysql5	Encrypt the password of MySQL accounts	MySQL database (Version 5.0 and later versions)
mssql	Encrypt the password of SQL Server accounts	Microsoft SQL Server
Des(unix)	Encrypt the password by DES, and save the value with Unix shadow format	Unix/Linux systems

Current cipher text attack methods; some wrong methods; our open-source samples

CIPHER TEXT ATTACK METHODS AND CURRENT SOLUTIONS

Attack Statistics

	Count							
	Number	Percent	Number	Percent	Number	Percent	Number	Percent
User	111775213	100.00%	18333811	16.40%	6428632	5.75%	31760979	28.42%
Passwords (after Elimination)	33060578	29.58%	5211738	28.43%	4037919	62.81%	13741971	43.27%
Passwords (pure number)	63214212	56.55%	9516238	51.91%	2893401	45.01%	19706359	62.05%
Passwords (with letters)	12141004	10.86%	1877018	10.24%	79365	1.23%	3168090	9.97%
Passwords(lowercase)	11655558	10.43%	1835567	10.01%	748554	11.64%	3063972	9.65%
Passwords(uppercase)	583986	0.52%	24077	0.13%	30396	0.47%	55492	0.17%
Passwords(user name)	4195863	3.75%	740388	4.04%	292662	4.55%	323500	1.02%
Passwords(part of user name)	3678526	3.29%	1255568	6.85%	188873	2.94%	574123	1.81%
Passwords(part of user name + characters)	1642673	1.47%	119229	0.65%	129390	2.01%	1010928	3.18%
Top 100 passwords	24282282	21.72%	3909054	21.32%	1704944	26.52%	7106500	22.37%
Weak passwords	1996150	1.79%	574459	3.13%	104128	1.62%	963493	3.03%
Passwords (with special characters)	3923457	3.51%	75107	0.41%	237675	3.70%	1448057	4.56%

High Frequency Match (Continued)

	Count							
Rank	Passwords	Number	Passwords	Number	Passwords	Number	Passwords	Number
1	123456	7516122	123456	1231988	123456789	470024	123456	2456830
2	123456789	1980576	111111	305060	12345678	425498	111111	615054
3	111111	1631232	123456789	291076	11111111	152692	000000	379674
4	123123	807392	123123	204428	dearbook	92106	123456789	367162
5	163.com	775834	000000	159910	00000000	69904	123123	218826
6	12345678	746450	5201314	115962	123123123	39972	123321	124066
7		645578	a123456	110982	1234567890	35580	5201314	117100
8	000000	641088	a321654	89102	88888888	30066	12345678	113562
9	5201314	557722	123321	53602	111111111	13990	666666	104840
10	0	516282	aaaaaa	43008	147258369	11930	111222tia	97720
11	123321	326256	7758521	42254	987654321	11106	888888	96958
12	11111111	291804	1314520	42108	aaaaaaaa	10918	1234567	86384
13	111222tian	265240	123456a	37366	1111111111	10290	654321	81724
14	a123456	264812	123	34168	66666666	10050	121212	70782
15	666666	244262	woaini	32690	a123456789	8870	789456	59442
16	1234567	229292	123123123	31916	11223344	8192	111222	58684
17	1314520	224386	12345678	30530	1qaz2wsx	7334	woaini	57888
18	7758521	210072	123456789	29194	xiazhili	7298	112233	55400
19	888888	201940	1234567	27616	789456123	7220	1314520	53262
20	123	193206	wangyut2	24722	password	7002	7758521	51322
21	1234567890	192456	112233	24304	87654321	6562	0	50748
22	654321	174456	11111111	23570	qqqqqqqq	6554	88888888	48484
23	woaini	161956	5211314	22442	000000000	6350	11111111	47726
24	112233	148884	qq123456	22312	qwertyuiop	6286	123456789	47188
25	123123123	146106	666666	22196	qq123456	6188	131313	45522

(Data Storage p13)

Plain text Password Match

Normal Speed

Size of data tested	692 bytes	51KB	550KB	27MB	371MB
algorithm/bit	ms/time	ms/time	ms/time	ms/time	ms/time
MD2/128	0.1453	10.2	110.62	5578.1	75550
MD4/128	0.0015	0.15	1.32	65.6	898
MD5/128	0.0023	0.23	2.03	103.1	1421
SHA/160	0.0039	0.31	2.8		
SHA1/160	0.0031	0.31	2.9		
SHA224/224	0.0109	0.93	9.(		
SHA256/256	0.0109	0.93	9.2		
SHA384/384	0.0242	1.63	17.1		
SHA512/512	0.025	1.63	16.8		
CRC32/32	0.0015	0.15	2.(		
CRC64/64	0.0046	0.16	2.7		

After GPU Acceleration

Hash Type	Average Speed (when attacking one hash on NVIDIA GTS250)
MD5	420 million p/s
MySQL	700 million p/s
MD4	590 million p/s
NTLM	550 million p/s
SHA-1	103 million p/s
SHA-256	65 million p/s
SHA-384	12.5 million p/s
SHA-512	12.3 million p/s
MySQL5	60 million p/s
LM	50 million p/s
Domain Cached Credentials	240 million p/s
md5(md5($pass))	165 million p/s
md5(sha1($pass))	70 million p/s
md5($pass.$salt)	173 million p/s
md5($salt.$pass)	230 million p/s
md5(md5($pass).$salt)	140 million p/s
md5(md5($salt).$pass)	225 million p/s
md5($salt.$pass.$salt)	235 million p/s
md5($salt.md5($pass))	85 million p/s
md5(md5($salt).md5($pass))	105 million p/s
md5(md5($pass).md5($salt))	100 million p/s
md5($username.'\0'.$pass)	250 million p/s
sha1(md5($pass))	66 million p/s
sha1($pass.$salt)	55 million p/s
sha1($username.$pass)	62 million p/s
sha1(strtolower($username).$pass)	62 million p/s
sha256(md5($pass))	45 million p/s
DES(Unix)	1.3 million p/s
MD5(Unix)	0.16 million p/s
MD5(APR)	0.16 million p/s
MD5(phpBB3)	0.2 million p/s
MD5(Wordpress)	0.05 million p/s
MD5(Half)	410 million p/s
MD5(Middle)	400 million p/s
MD5(Unicode)	375 million p/s
RAdmin v2.x	210 million p/s
GOST R 34.11-94	0.2 million p/s

引自http://www.insidepro.com/eng/egb.shtml

Some Wrong Methods

- Use standard hash algorithms
- Use several hash values
- Use non-unidirectional algorithms
- Add salt
- Design algorithms

Uniformity test of Hash function

Test algorithm: CRC64

Chi square test: Degree of confidence: 0.99 Number of samples: 512

Number of data tested: 30 Test number: 40 Chi square value: 588.7

Chi square value of tested data:

CRC64		MD5	fnv_1_64	DJBHash	BKDRHash	SDBMhash	md4	crc32
599.0127		503.5691	890.0062	301418.4	301930.9	300902.3	538.1734	518.7796
507.3101		450.5395	881.1998	301428.9	301906.3	300841.4	501.8385	547.8741
515.3997		508.8358	831.2286	301345.5	301863.8	300960.1	510.5801	510.3275
486.4307		487.3011	817.2885	301343.5	301980.8	300800.6	539.9712	541.7038
449.7986		520.8506	889.6751	301393.8	301866.9	300899.8	497.0906	533.5825
472.4702		501.917	817.6708	301418.7	301778.4	300915.9	504.7433	520.4309
548.4646		488.6699	826.3543	301318	302203.2	301042.7	464.7765	503.9275
538.6411		586.6532	827.2452	301280.5	301998.2	300861.8	490.4516	474.5318
494.3701		498.2955	875.4415	301309	301760.5	300856	506.6717	563.6813
530.5788		497.0462	840.4753	301307.3	302052.2	300872.6	498.5071	478.5766
520.0521		532.6541	860.5901	301432.2	301975.6	300858.8	558.1898	459.2162
514.403		496.7936	830.3753	301326.5	301907.4	301076.2	523.2947	499.4048
535.5076		456.2091	853.5313	301407.1	302081.4	300847.3	510.1773	502.6577
526.1722		516.8401	810.0454	301368.9	301835.3	300926.8	493.5066	477.283
517.8286		492.1958	787.8076	301311.9	301925.8	300914.6	589.2679	522.1171
479.4607		492.6293	852.8486	301503.5	301776.6	300895.5	532.6711	553.7553
551.2977		477.0577	807.5162	301370.7	302176.1	300918.1	548.0209	506.9756
517.8027		492.3017	851.6038	301303.4	301724.2	300954.5	507.194	496.2081
464.1894		497.3158	849.4114	301426.2	302134.3	300927.1	549.3385	524.2812
499.7257		518.5946	926.307	301358.9	301739.1	300892.5	497.8449	503.9753
512.8055		463.0733	835.683	301289.5	301994.5	301067.2	498.0329	522.1786
527.9949		502.1457	787.2512	301342.9	302079.1	300831.5	448.099	496.739
516.3349		547.8844	777.6905	301319.1	301809.9	300851.7	451.3109	511.0238
491.8716		451.7079	911.4419	301342.1	302007.8	300747.1	499.654	507.1462
493.155		509.3649	796.7539	301388.8	301939.2	300893.9	504.2890	506.6445
517.83		476.9553	816.326	301415.2	301856.7	301062.9	561.6845	514.7307
503.5179		461.3495	820.7667	301397.7	301919.6	300847.7	548.6251	548.4407
508.4706		578.7542	839.3967	301327.3	301964	300869.9	466.722	569.738
506.5455		477.3690	787.2239	301452.5	301853.4	300852.4	491.2333	477.5083
534.4905		499.3126	887.0332	301386	301817.1	300964.3	519.3899	607.2764
514.3518		426.6564	841.2706	301465.1	302177.2	300884.9	492.5611	533.8453
504.9344		516.509	850.9747	301375.6	301743	300947.9	548.4783	459.1479
471.8524		491.2947	878.9641	301418.6	302061.8	300813.8	534.3505	509.4775
467.5311		547.9663	860.317	301440	301868.4	300930.6	482.0685	528.483
498.9431		502.5212	865.833	301280.7	301768.4	300950.4	540.2796	535.2823
523.619		549.4511	789.4835	301309	302200.7	300871.8	446.8497	548.1813
506.7437		517.0074	783.8891	301263	301803.7	300781.6	487.1885	543.1774
498.3057		514.5805	912.4396	301237.6	302000.8	300888	494.1813	562.6061
540.4467		532.8316	835.2597	301432.4	301840.1	300815	472.3644	483.6727
504.5655		483.8537	814.4111	301327.3	301861	300978.7	518.5056	536.6613
Mean value								
510.1306		501.169	840.4757	301361.1	301930.9	300902.3	507.1795	518.7796
Conclusion								
Not negative		Not negative	Negative	Negative	Negative	Negative	Not negative	Not negative

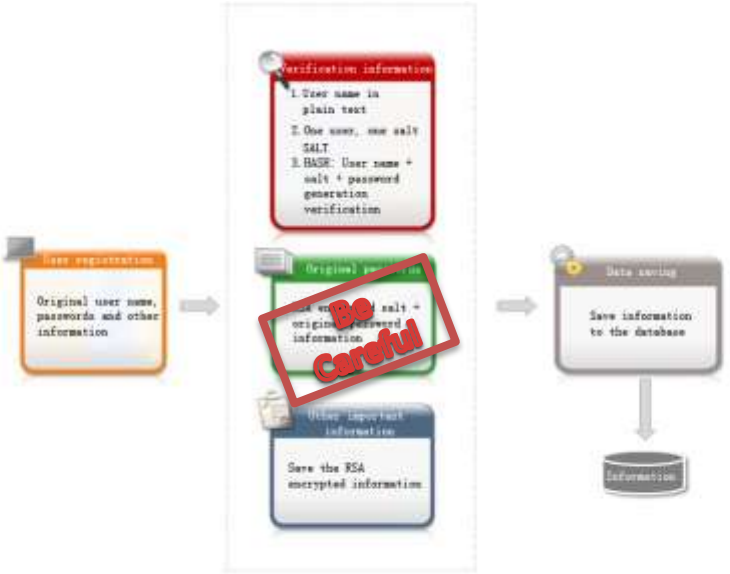
Antiy Password Mixer

Introduction

- Algorithm
  - RSA, SHA256
- User/Salt
  - Accounts
  - The salt table
  - UID and registration time
- Provide restoration mode

Resources

- Open source
  - http://code.google.com/p/password-mixer/



Not every security product is useful. We should recognize the fake ones.

FIND SUITABLE SECURITY PRODUCTS

Design Slow Hash

Questions

- Fake slow hash
  - Another algorithm
- User's experience
- DDoS attacks

Perspective

- It's difficult to design a slow hash algorithm. Such a algorithm is of low payload for X86 CPUs. Moreover, it is difficult to bypass certain software and hardware.

Biometric Recognition

Question

- It is not what we are talking about here.
- It’s not renewable.

Perspective

- The identity recognition technology based on the Internet is widely used; it is disastrous since users’ information might be stolen.
- Some websites advocate the identity recognition technologies. They might be driven by economic interests.

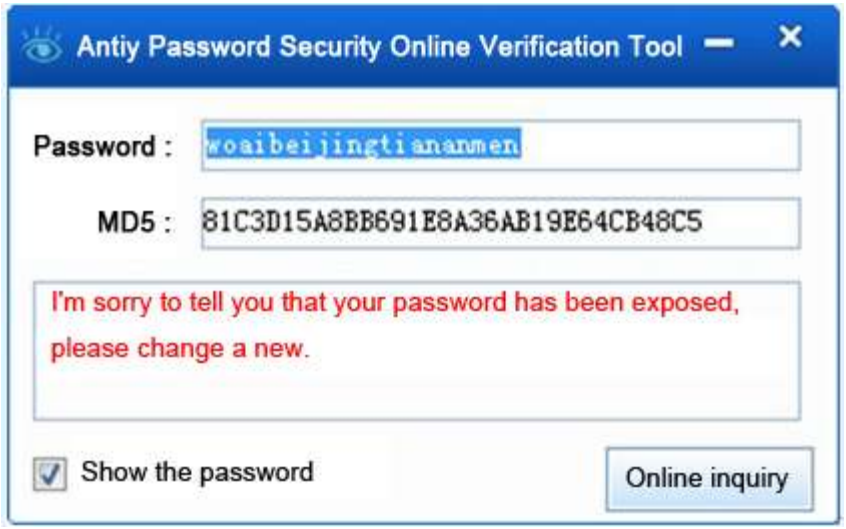
Zhang's Hypothesis

- Any Web/DB targeted logon strategies must be based on the following conditions.
  - open source algorithms
  - Rapid hash algorithm
  - The rainbow table is only limited by storage.

Avoid Using Frequently Used Passwords

Don’t use Frequently Used Passwords

- Check the passwords that are used for several accounts;
- High frequency password and leaked passwords



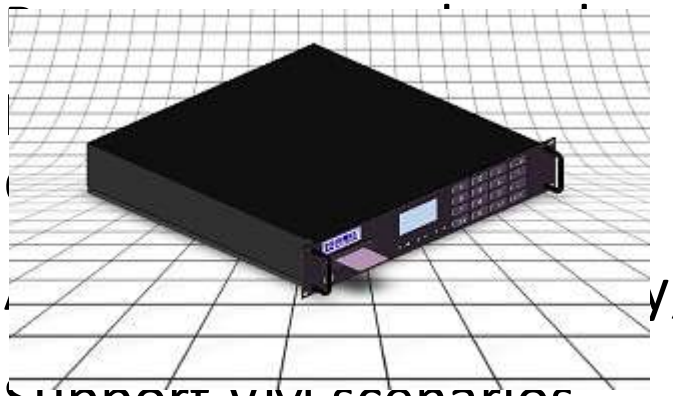
Dynamic Balance

- Microsoft Hotmail
  - Frequency balance
  - Shortcomings

Encryption Hardware Used for WEB/DB Scenarios

Idea

Design

- 
- ;
- Support ViVi scenarios

Operation and Maintenance

The following problem must be settled

- Counter password and the online passwords should be different.

Try the following one

- Protect records via records
  - Create lots of bot users;
  - Misled by algorithms
  - Fake table

The pseudo propositions …

EXTRA TOPICS

Some Pseudo Propositions

- Unidirectional security?
  - Cracking situation（MD5 security is not relevant with this event)
  - intensity
- Dual factor security?
  - Financial institutes should bear the responsibilities.

References

- Password related article:
  - http://blog.csdn.net/antiy_seak
- Antiy Password Mixer
  - http://code.google.com/p/password-mixer/
- Speed of GPU acceleration related algorithms
  - http://www.insidepro.com/eng/egb.shtml

Thank You

www.antiy.com

seak@antiy.com

http://Weibo.com/seak