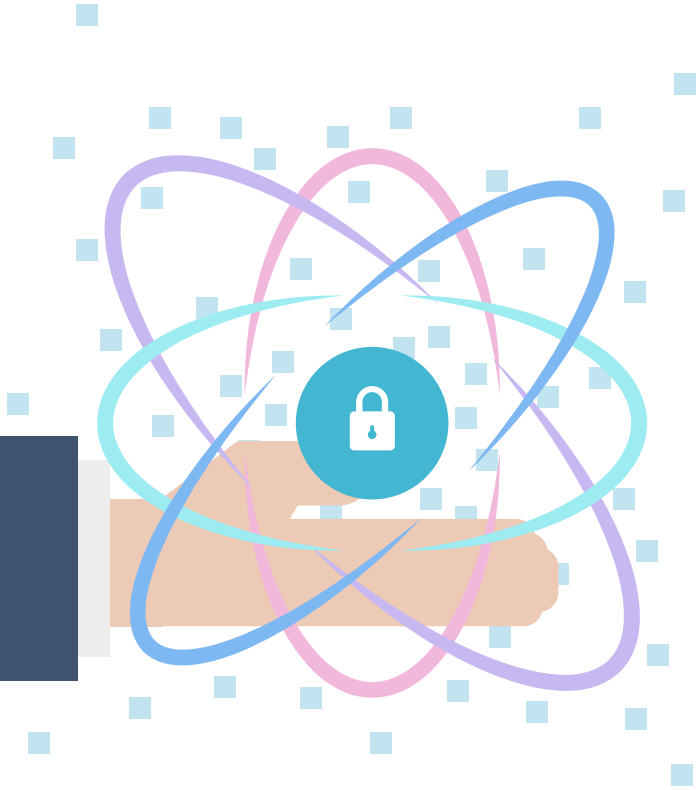
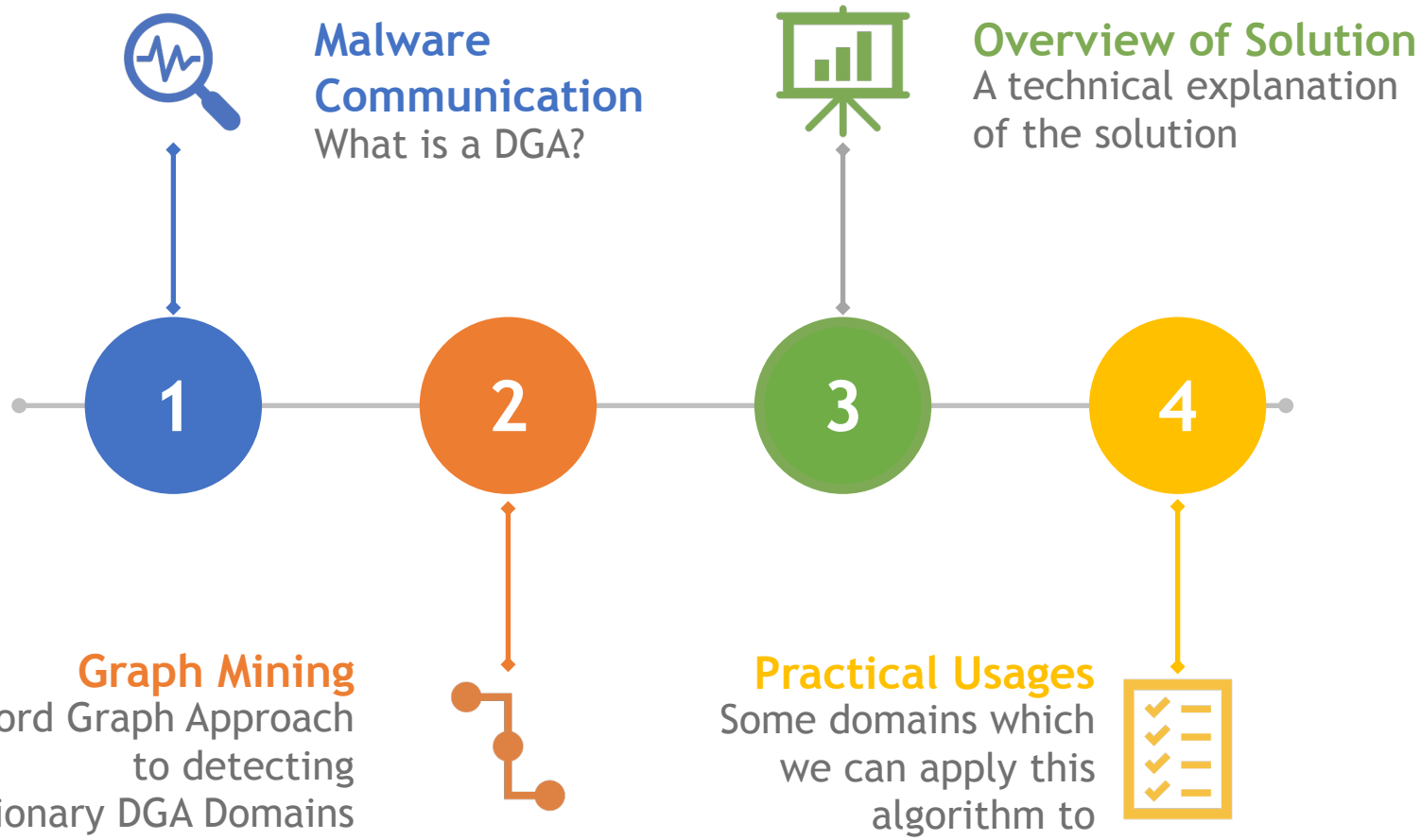
Graph Based Machine Learning



Detecting Wordlist Domain Generation Algorithms

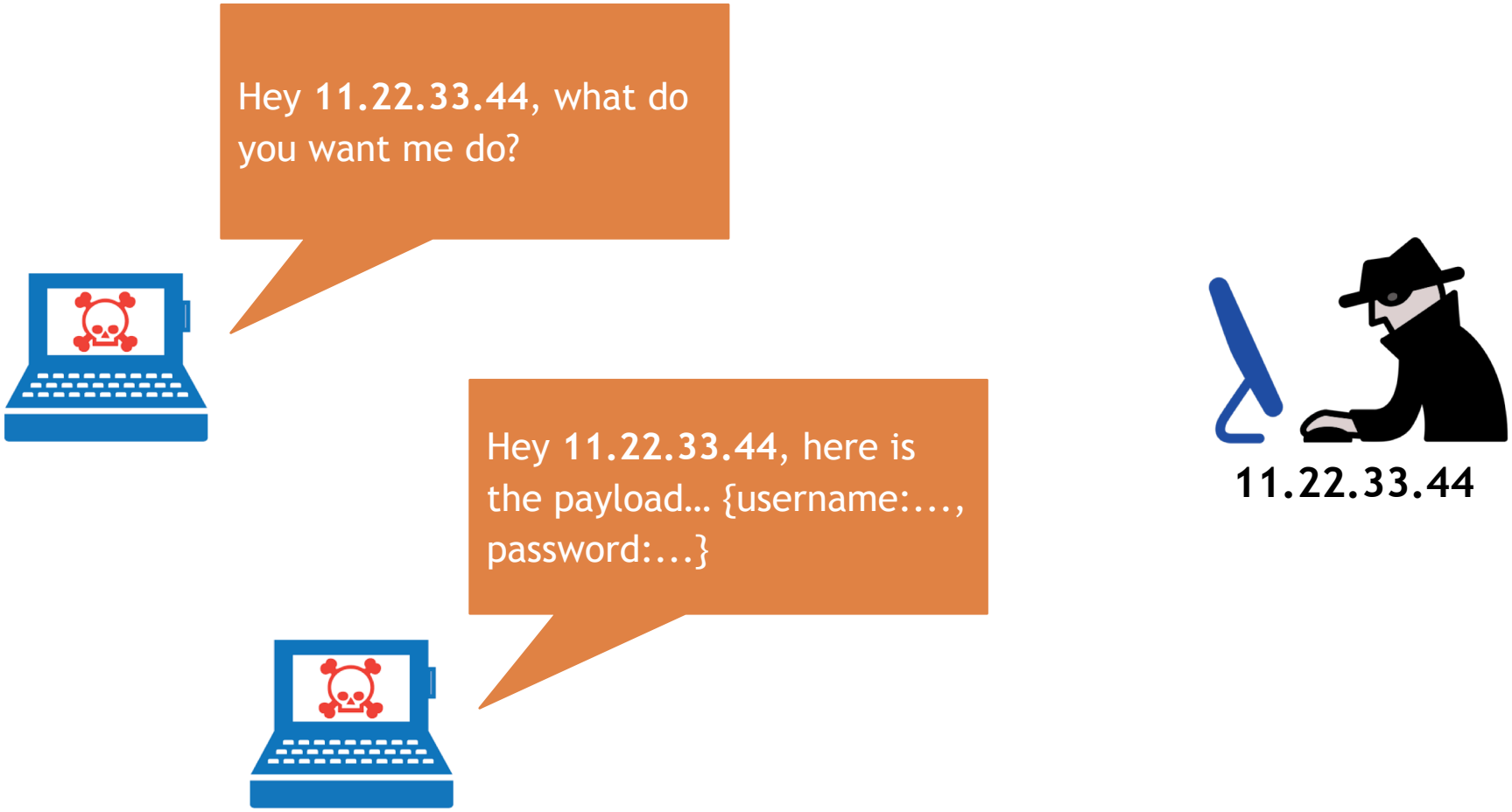
Sng Yu Feng Chester @chestersng
Lam Fhong Sheng @chrispooh007

Roadmap


Malware Communication
What is a DGA?
Overview of Solution
A technical explanation of the solution
1
2
3
4
Graph Mining
Word Graph Approach to detecting Dictionary DGA Domains
Practical Usages
Some domains which we can apply this algorithm to

Attacker's POV

Hardcoded IP Address



Author writes DGA

```
def dga(date, magic, number):
    seed = date.year + date.month + date.day + magic
    r = Rand(seed)
    if i == 0x33:
        r = Rand(magic)
    v1 = r.rand()
    ra = []
    for i in range(10):
        ra.append(v1 % 10)
        v1 //= 10

    domain = ""
    for x in ra:
        domain += LETTERS[x]

    if ra[0] < len(TLDS):
        tld = TLDS[ra[0]]
    else:
        tld = DEFAULTTLD

    domain += tld
    yield domain
```

Author writes DGA

```
def dga(date, magic, number):
    seed = date.year + date.month + date.day + magic
    r = Rand(seed)

    for i in range(1, number):
        if i == 0x33:
            r = Rand(magic)
        v1 = r.rand()
        ra = []
        for i in range(10):
            ra.append(v1 % 10)
            v1 //= 10

        domain = ""
        for x in ra:
            domain += LETTERS[x]

        if ra[0] < len(TLDS):
            tld = TLDS[ra[0]]
        else:
            tld = DEFAULTTLD

        domain += tld
        yield domain
```


```
def dga(date, magic, number):
    seed = date.year + date.month + date.day + magic
    r = Rand(seed)

    for i in range(1, number):
        if i == 0x33:
            r = Rand(magic)
        v1 = r.rand()
        ra = []
        for i in range(10):
            ra.append(v1 % 10)
            v1 //= 10

        domain = ""
        for x in ra:
            domain += LETTERS[x]

        if ra[0] < len(TLDS):
            tld = TLDS[ra[0]]
        else:
            tld = DEFAULTTLD

        domain += tld
        yield domain
```



```
def dga(date, magic, number):
    seed = date.year + date.month + date.day + magic
    r = Rand(seed)

    for i in range(1, number):
        if i == 0x33:
            r = Rand(magic)
        v1 = r.rand()
        ra = []
        for i in range(10):
            ra.append(v1 % 10)
            v1 //= 10

        domain = ""
        for x in ra:
            domain += LETTERS[x]

        if ra[0] < len(TLDS):
            tld = TLDS[ra[0]]
        else:
            tld = DEFAULTTLD

        domain += tld
        yield domain
```

Step 1

Run DGA in advance with time seed of tomorrow 7p.m.

Produces:

a9djc92c.com

d02kd0x.net

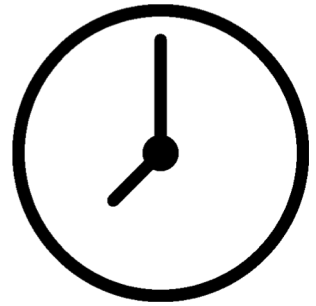
19cjsodos.org

...

Registers one of them:

19cjsodos.org

Step 2



Run DGA with time seed <u>now</u>

Produces:

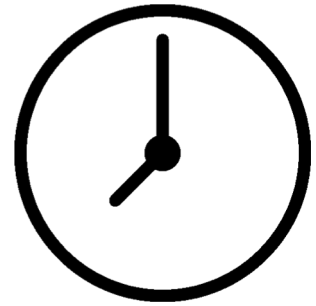
a9djc92c.com

d02kd0x.net

19cjsodos.org

…

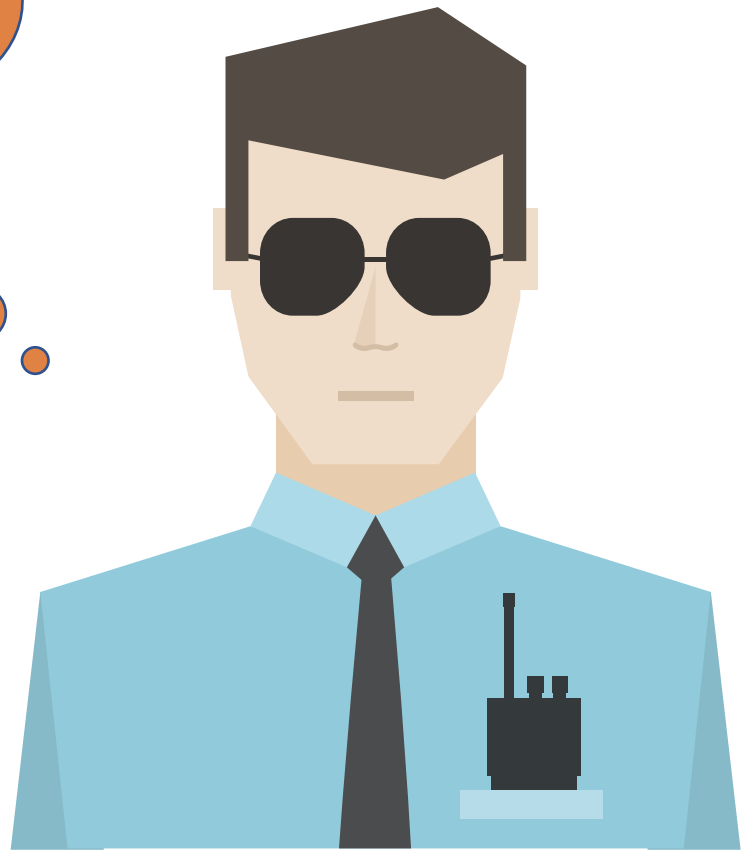
Step 3



Attempts to communicate with all of them:
a9djc92c.com
d02kd0x.net
19cjsodos.org
...
Successfully Established Connection with:
19cjsodos.org

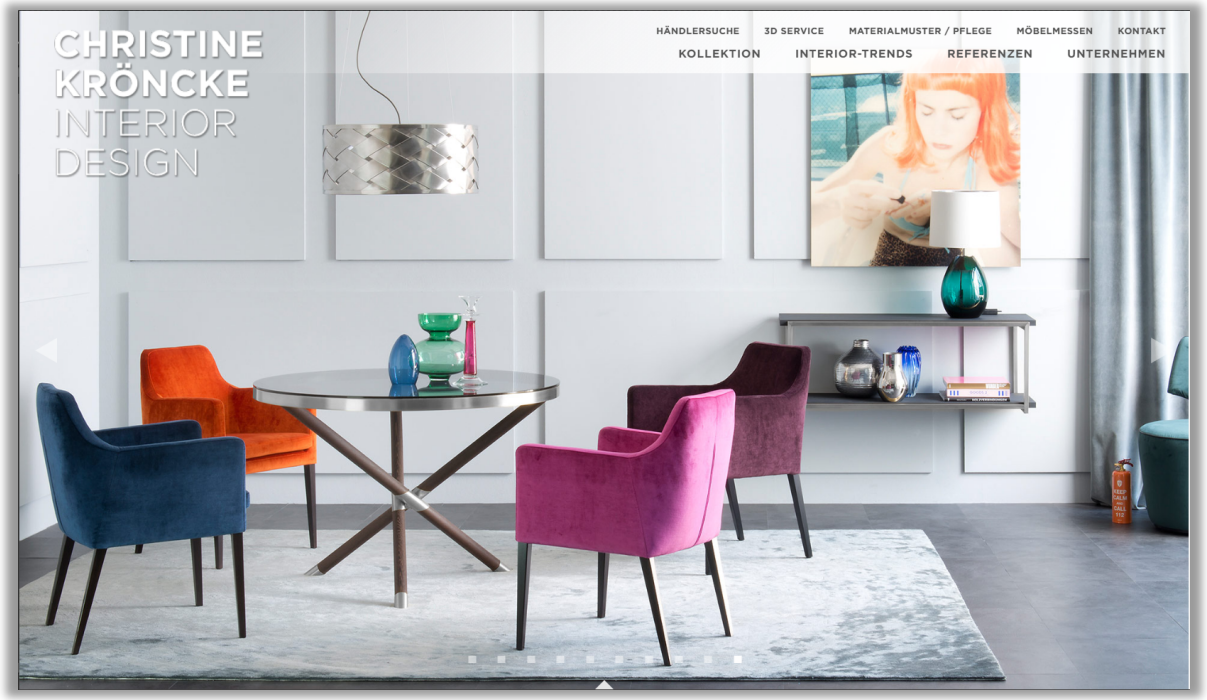
Defender's POV

Can I **identify** DGA-generated domains?

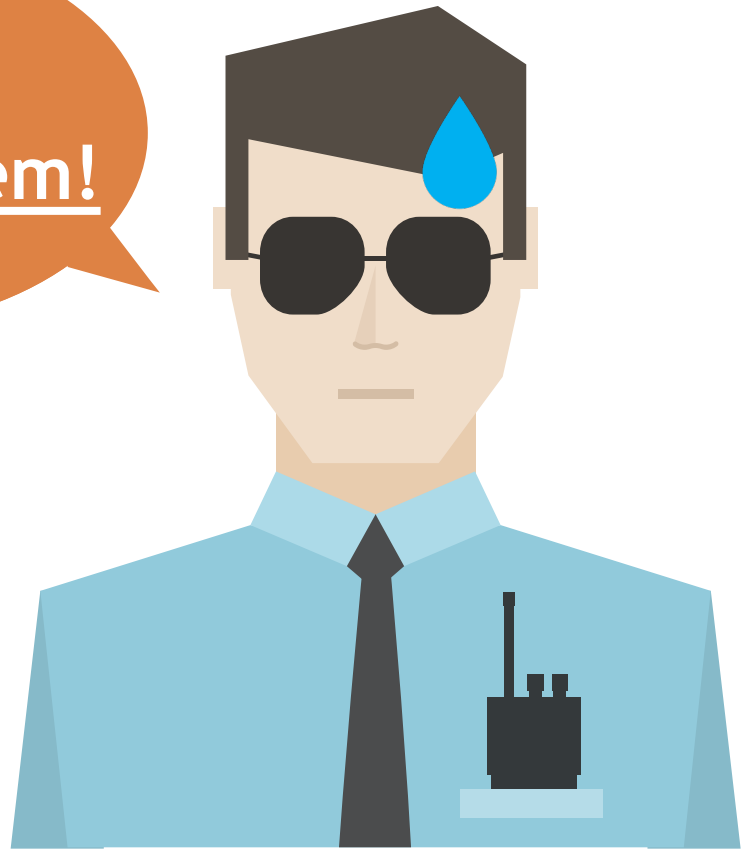


Which is the DGA Domain?

Which is the DGA Domain?



This is a hard problem!



Not just one type of DGA…

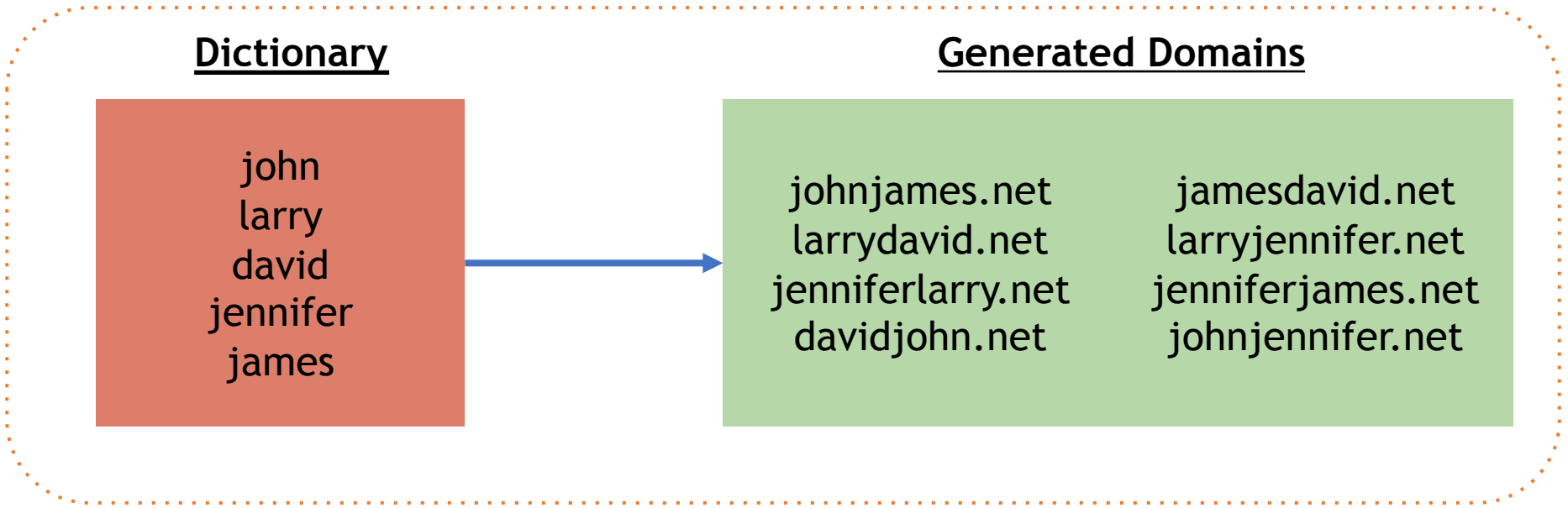
Types	Example Family	Examples Domains
Arithmetic (A)	DirCrypt	wejcqzbosbczzlnikyvt.com, muiccxbvkvjb.com, tqwpmpwckhidiss.com, gzredieexn.com, ghhcwldtj.com
Wordlist (W)	Matsnu	accident-require.com paintfinance.com lawyersit-direction.com troublepace-summer.com
Hashing (H)	Bamital	cd8f66549913a78c5a8004c82bcf6b01.info aa24603b0defd57ebfef34befde16370.cz.cc 5e6efdd674c134ddb2a7a2e3c603cc14.org
Permutation	“Explosive” malware from “VolatileCedar"	explorerdotnt.info dotnetexplorer.info dotntexplorere.info xploreredotnet.info

2018 Research on DDGA Detection

Observation 1

Malware generate their domains using a **fixed-size** dictionary. Normally, these dictionaries are **small** (hundreds of words) to reduce payload size.

Word will be **reused**, since the size of the dictionary is small.

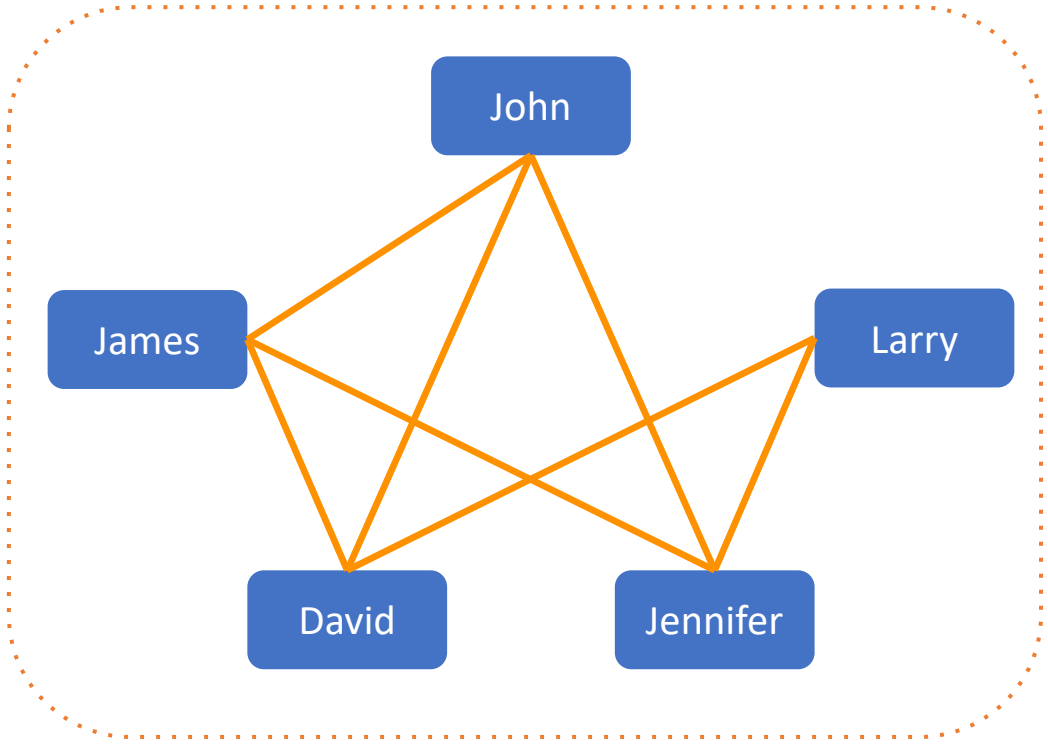


Observation 2

We can make use of **graph theory** to form a solution for this problem.

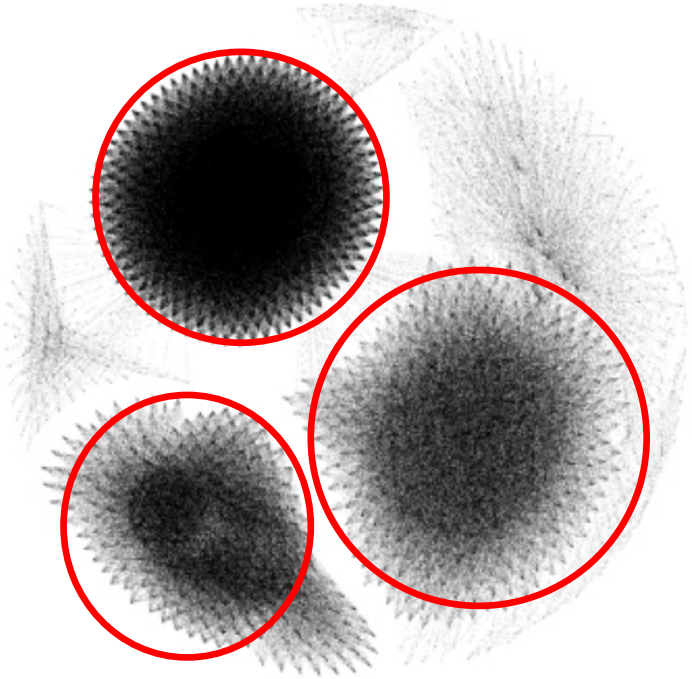
1. Let the words in the dictionary be **nodes**
2. Two nodes are linked if they **appear in the same domain.**

jamesdavid.net
larryjennifer.net
jenniferjames.net
johnjennifer.net

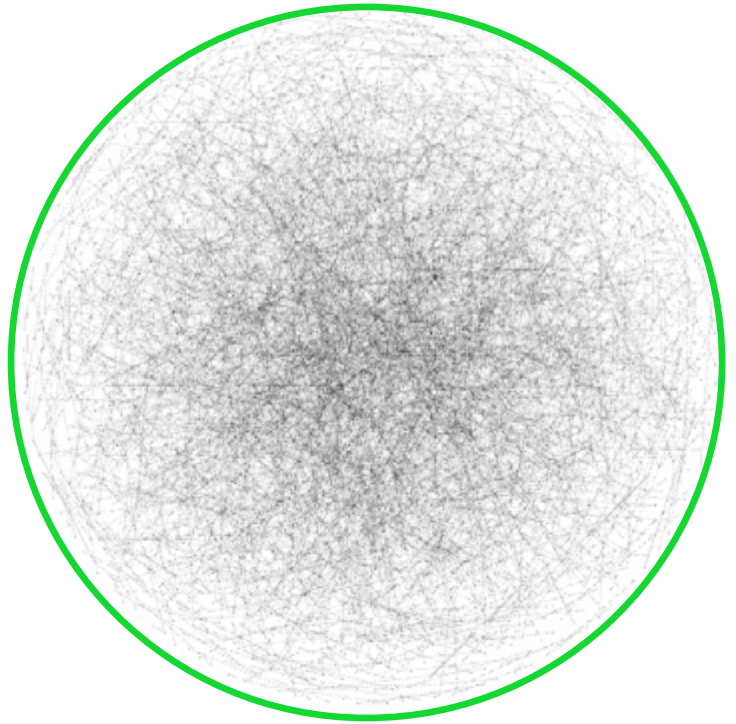


Observation 3

Dictionary-generated domains cluster **differently** from benign domains.



Dictionary Generated Clusters



Benign Domains Clusters

Components of the Solution

Word Breaking Algorithm

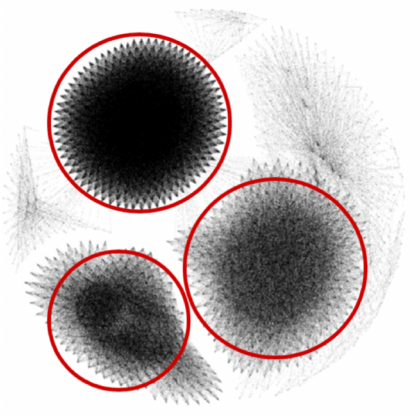
doghouse.net

dog house

Break domains up into individual words

Implemented a new algorithm to increase **efficiency**

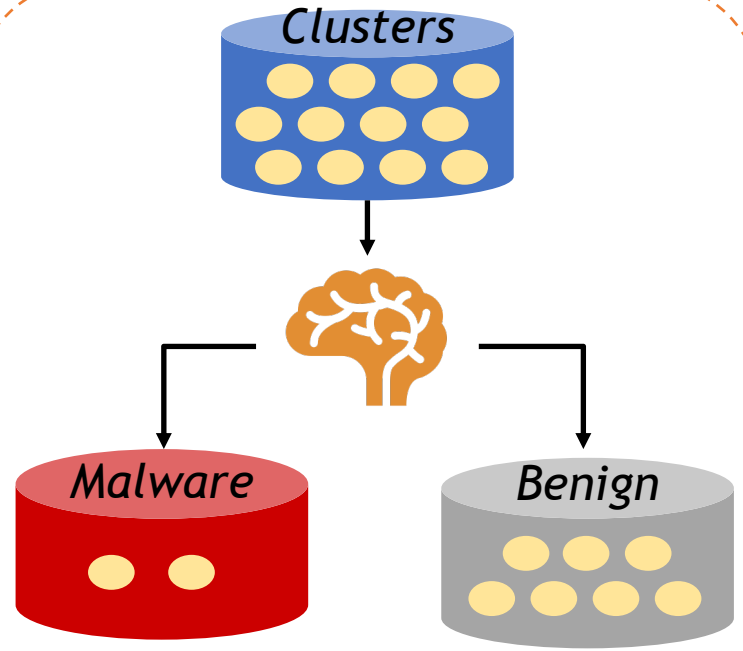
Graph Library



Create a **word graph** and **extract clusters**

Used an additional graph library to provide **clear** visualisation

Artificial Intelligence



Used a larger dataset to detect **more** families of malware

Repeated words!

christianasheenagh.net
catharinesheenagh.net
christianasherilyn.net
clevelandsheridan.net
mastermansherilyn.net
sebastiansherilyn.net
grenvilleshavonne.net
clevelandshakesheave.net
sebastiansheenagh.net
christophershavonne.net

grenvilleshakesheave.net
sebastiansheridan.net
clevelandsheenagh.net
grenvillesherilyn.net
mastermansheridan.net
grenvillesheenagh.net
christophershakesheave.net
sebastianshakesheave.net
christianasheridan.net
catharinesherilyn.net

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net

Word	Domains

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net
tomj	tomjack.net
ack	tomjack.net

Word	Domains

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net

Word	Domains

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net
jane	jilljane.net

Word	Domains

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net
jane	jilljane.net
jillj	jilljane.net
ane	jilljane.net

Word	Domains

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net
jane	jilljane.net
jillj	jilljane.net
ane	jilljane.net
jac	jackjill.net

Word	Domains
kjill	jackjill.net

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net, jackjill.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net, jackjill.net
jane	jilljane.net
jillj	jilljane.net
ane	jilljane.net
jac	jackjill.net

Word	Domains
kjill	jackjill.net

Wordbreaking Algorithm

k = 3

tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net
jack	tomjack.net, jackjill.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net, jackjill.net
jane	jilljane.net
jillj	jilljane.net
ane	jilljane.net
jac	jackjill.net

Word	Domains
kjill	jackjill.net
jackj	jackjill.net
ill	jackjill.net

Wordbreaking Algorithm

k = 3

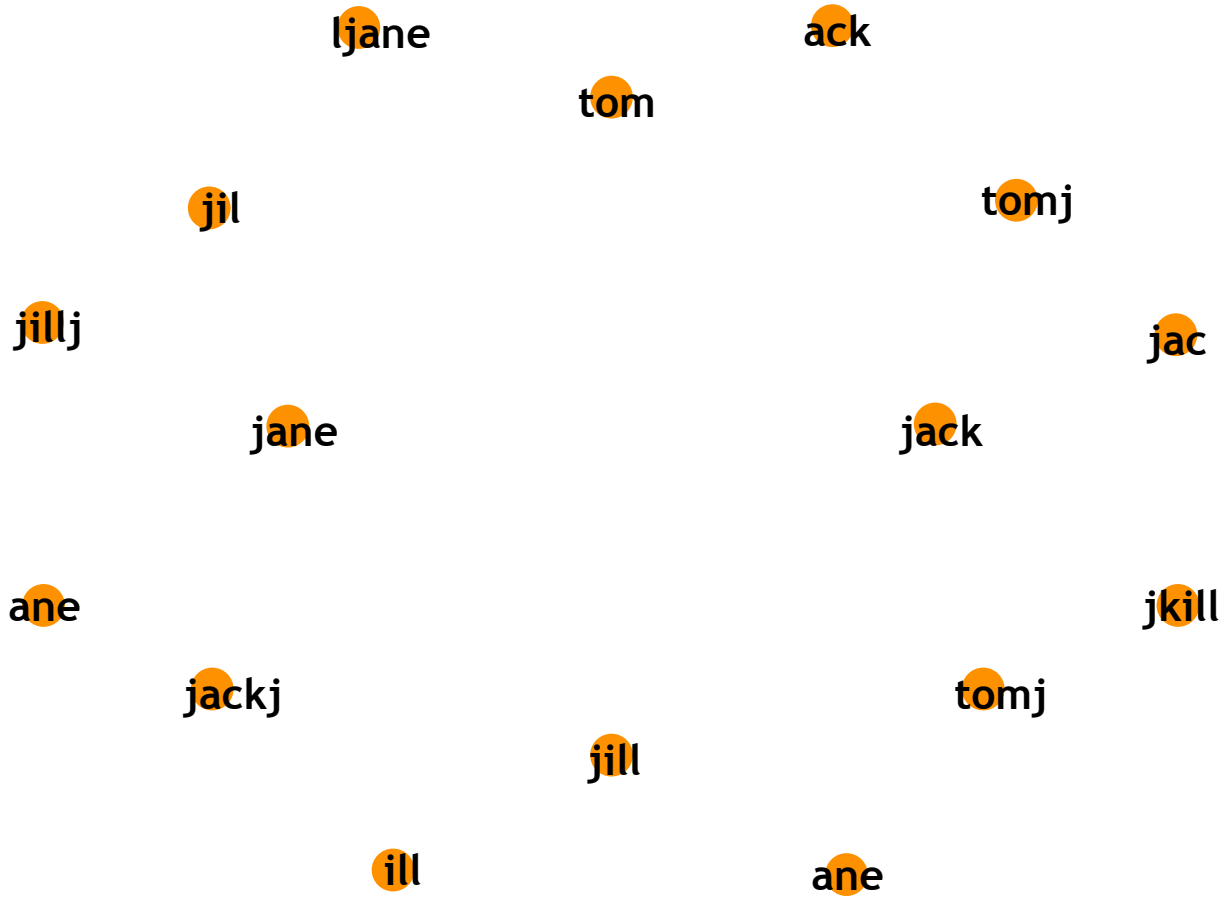
tomjack.net jilljane.net jackjill.net tomjane.net

Word	Domains
tom	tomjack.net, tomjane.net
jack	tomjack.net, jackjill.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net, jackjill.net
jane	jilljane.net, tomjane.net
jillj	jilljane.net
ane	jilljane.net
jac	jackjill.net

Word	Domains
kjill	jackjill.net
jackj	jackjill.net
ill	jackjill.net
tomj	tomjane.net
ane	tomjane.net

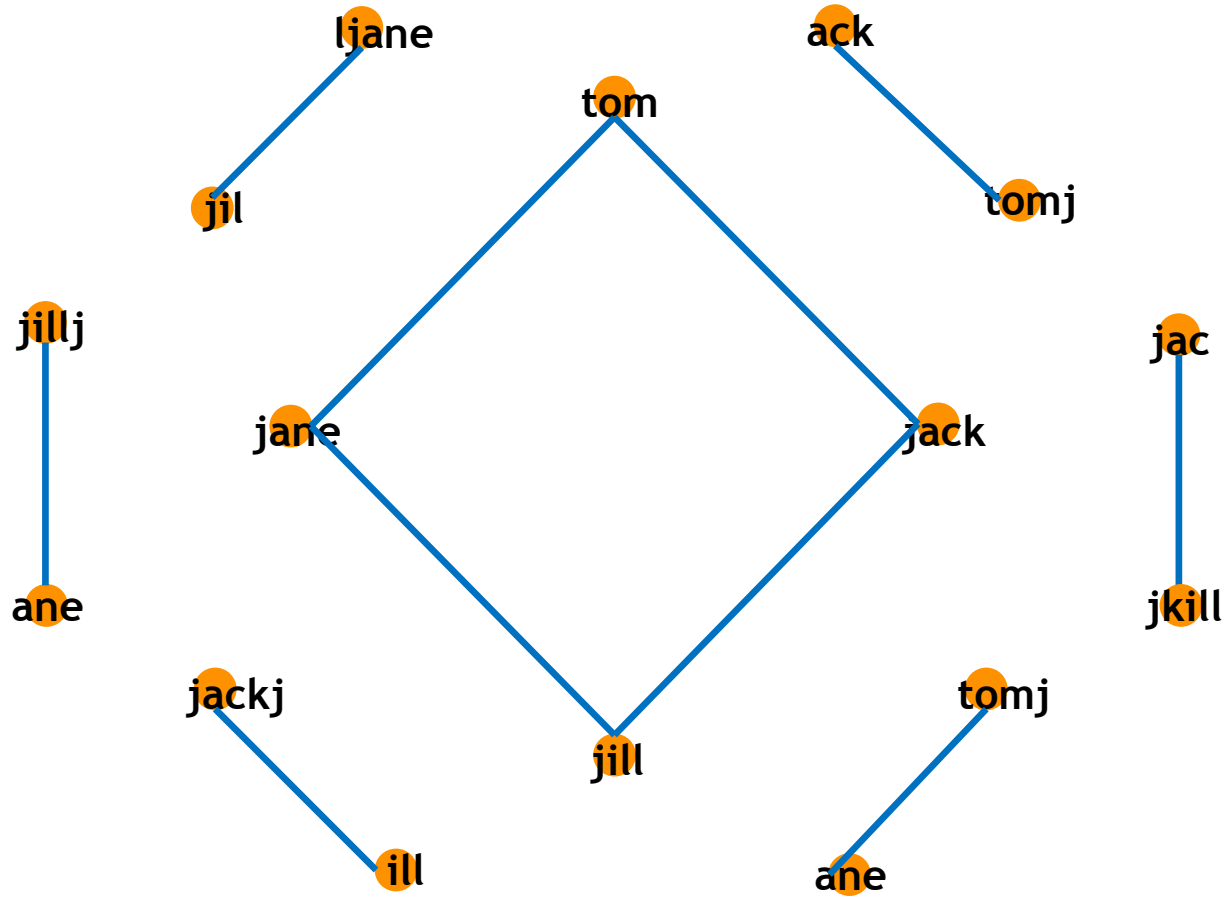
Graphing Algorithm

Word	Domains
tom	tomjack.net, tomjane.net
jack	tomjack.net, jackjill.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net, jackjill.net
jane	jilljane.net, tomjane.net
jillj	jilljane.net
ane	jilljane.net
jac	jackjill.net
kjill	jackjill.net
jackj	jackjill.net
ill	jackjill.net
tomj	tomjane.net
ane	tomjane.net

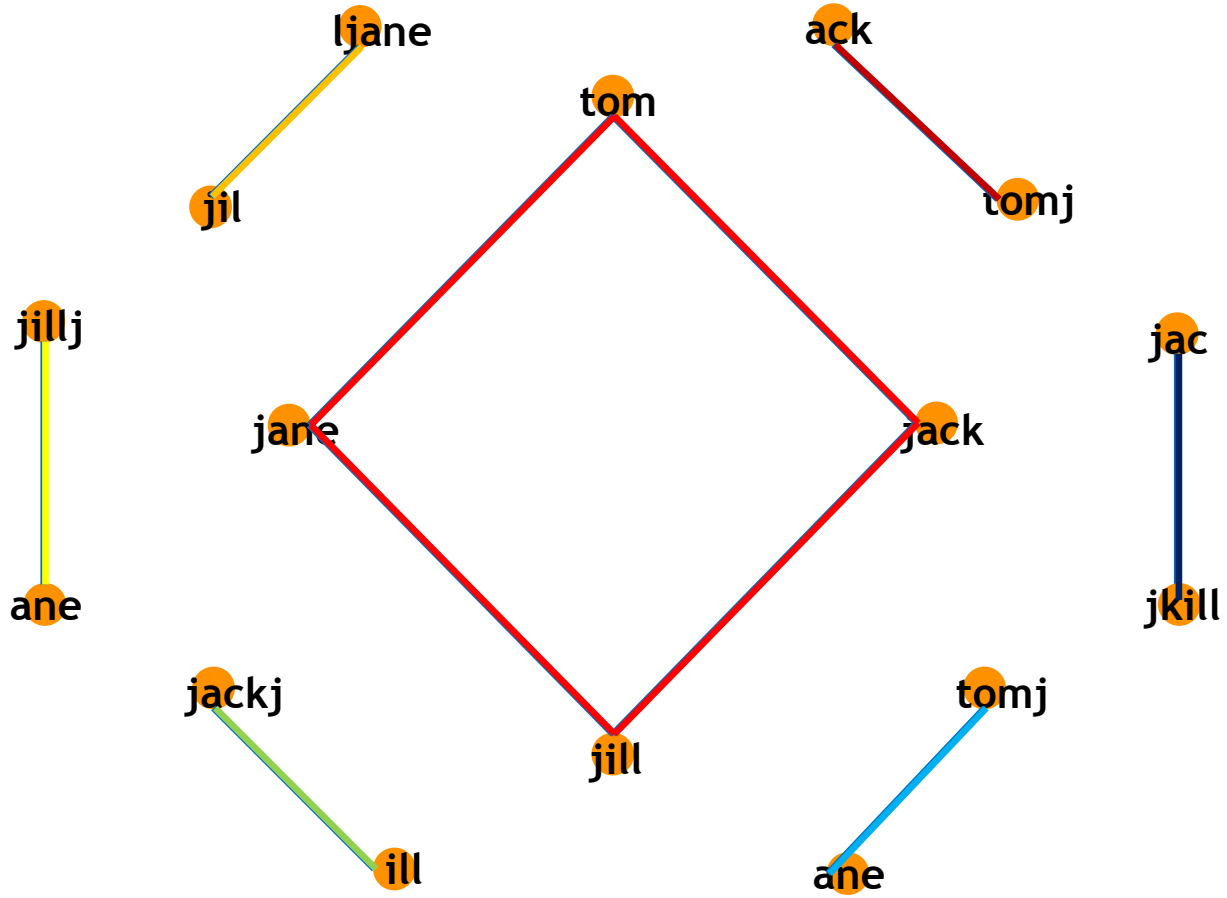


Graphing Algorithm

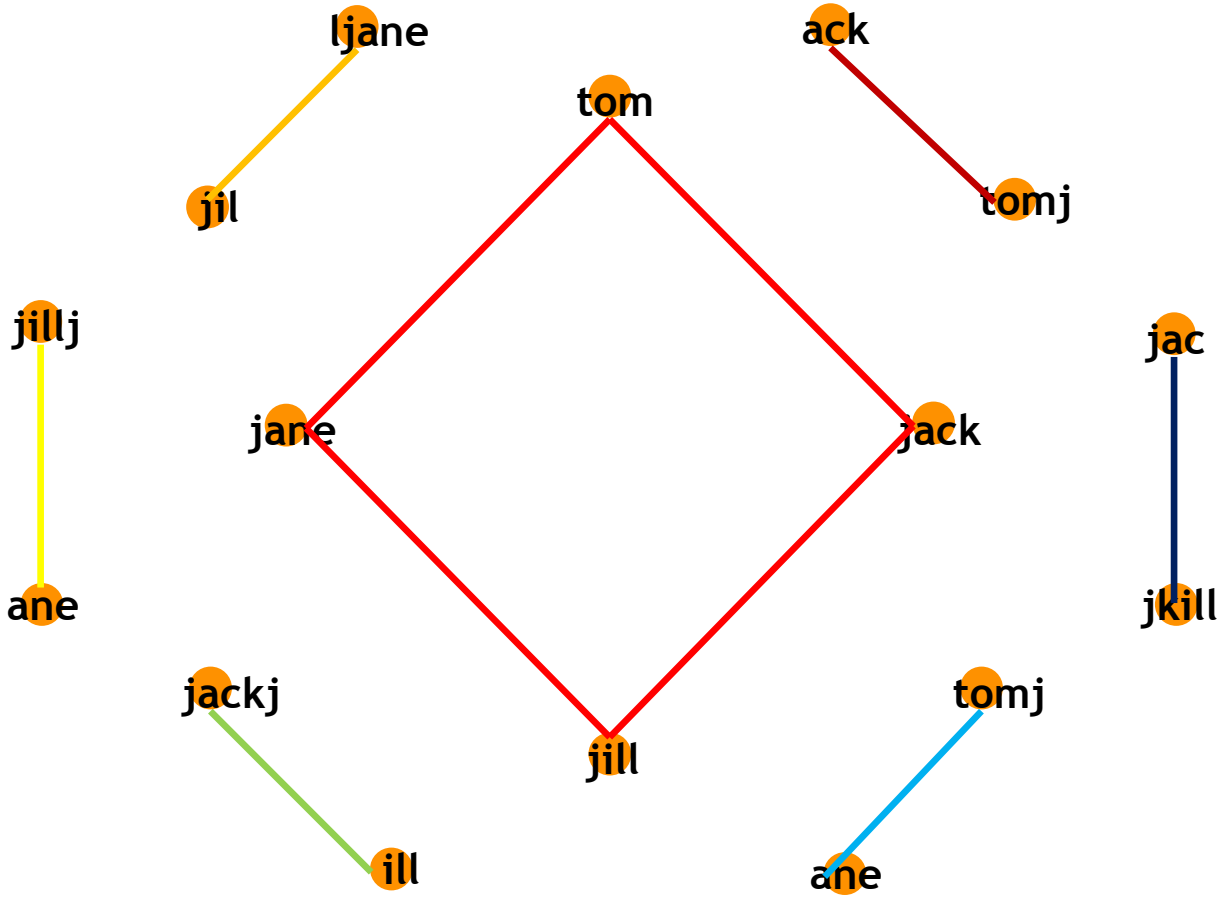
Word	Domains
tom	tomjack.net, tomjane.net
jack	tomjack.net, jackjill.net
tomj	tomjack.net
ack	tomjack.net
jil	jilljane.net
ljane	jilljane.net
jill	jilljane.net, jackjill.net
jane	jilljane.net, tomjane.net
jillj	jilljane.net
ane	jilljane.net
jac	jackjill.net
kjill	jackjill.net
jackj	jackjill.net
ill	jackjill.net
tomj	tomjane.net
ane	tomjane.net



Graphing Algorithm



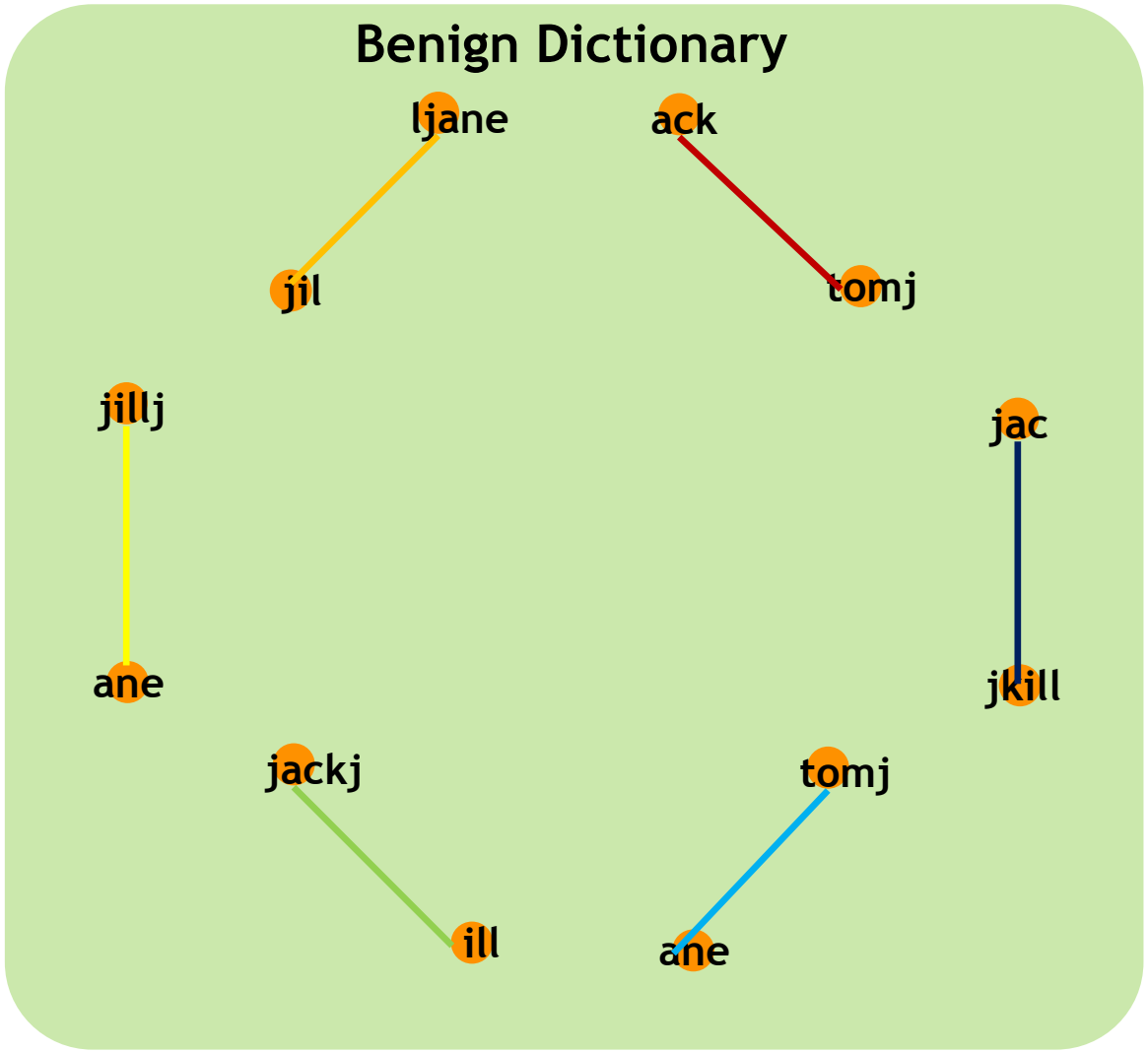
Artificial Intelligence

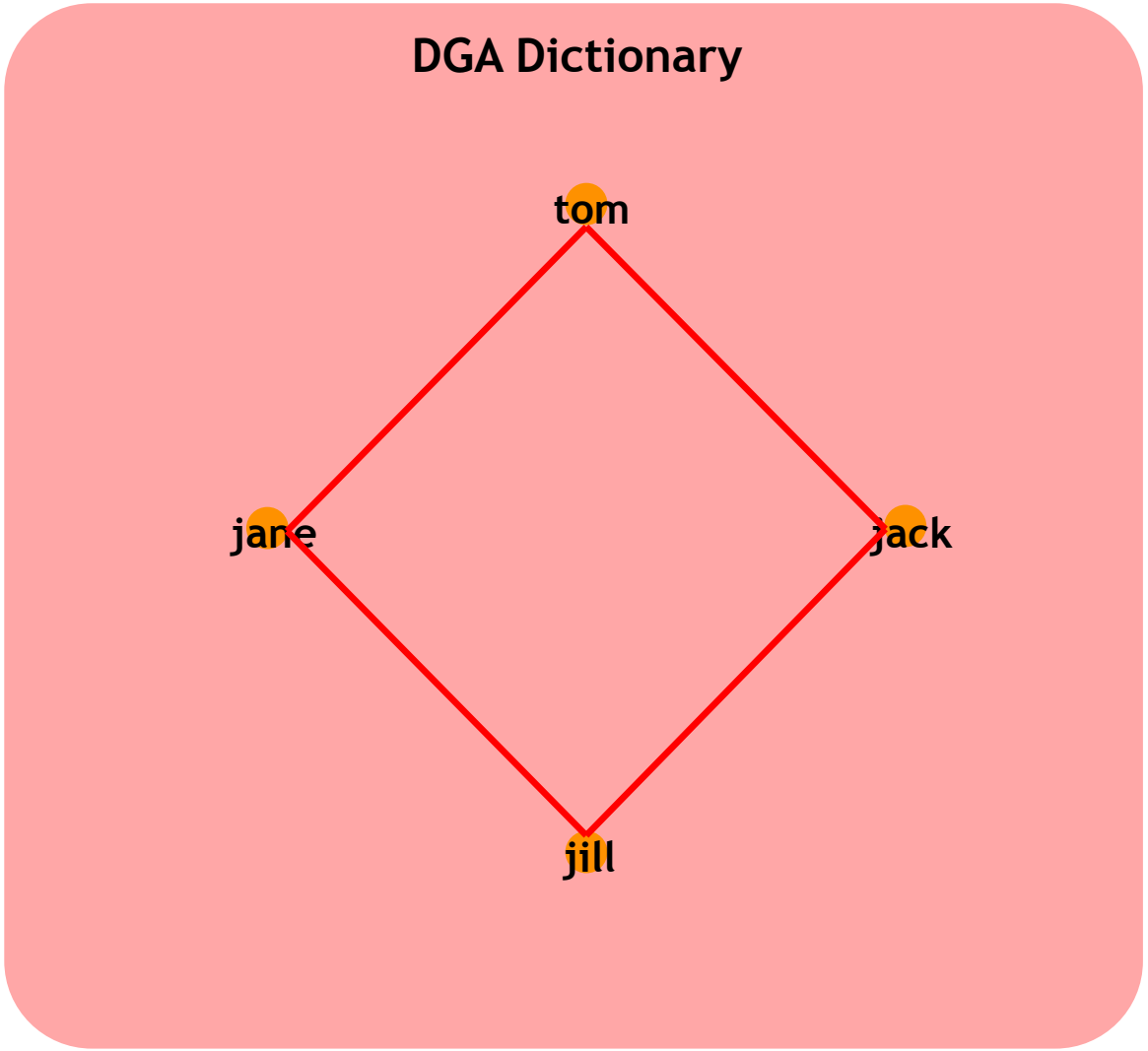


Feature Vector
Average Node Degree
Maximum Node Degree
Minimum Node Degree
Cardinality of Cycle Basis
Average Cycles Per Node
Average Clustering Per Node

Features that suggest **clustering**

Classification





Running on REAL DATA

How to mitigate BKDR_BAYROB using TrendMicro Products

Updated: 24 Nov 2016 **Product/Version:** Deep Discovery 3.0 , **Platform:** N/A N/A

SUMMARY

The BAYROB/NIVDORT malware arrives via spam email as an attachment and goes through two stages of infection. The first stage uses Domain Generating Algorithm (DGA) to download its updated copy that contains backdoor capability. The second stage of infection, downloads a bitcoin-miner on an infected machine.

BAYROB was first seen in 2007 stealing only eBay accounts until it evolved and appeared again in 2015 with its backdoor and Anti-AV capabilities. A resurgence in 2016 caused high volume of infections due to its mass mailing capabilities.

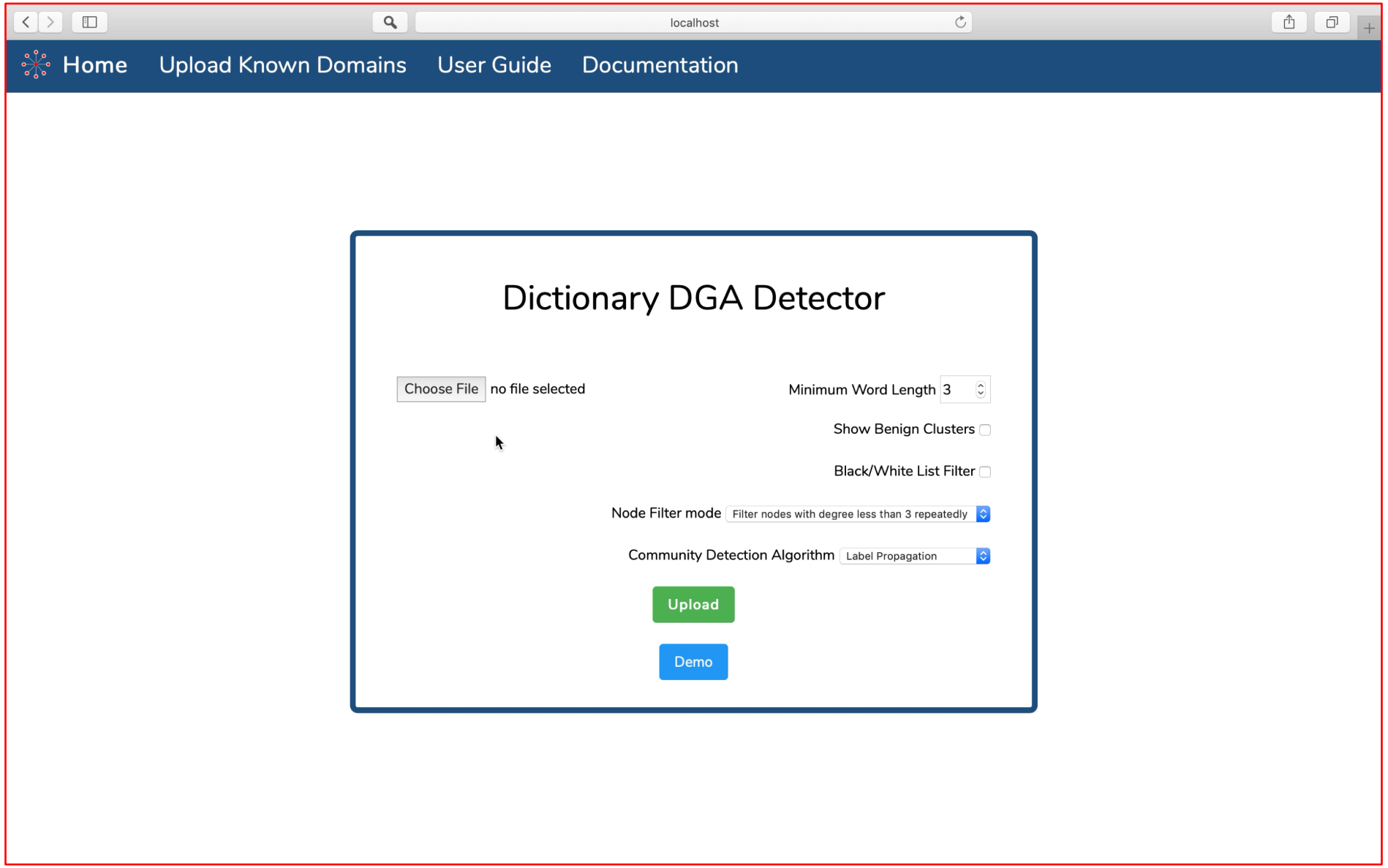
Below is a summary profile of this threat throughout the years of observation:

	2007	2015	2016
TARGET	eBay Accounts	Harvest bitcoin via coinminer	Still being determined by looking for live C&C
URL	Fixed malicious URL	Uses DGA	Uses DGA
STEALTH MECHANISM	Uses Kodak Viewer Express	Uses fake MP3 extensions and error messages	Uses names as filenames (jewell.exe) and error messages
CAPABILITY	Information theft	Information theft, backdoor capabilities, may download other malware with other functionalities, clicker capabilities	Information theft, backdoor capabilities, may download other malware with other functionalities, mass mailing capabilities
DEFENSE	None	Disables AV via registry, disables firewall, terminates AV application including watchdog	Disables AV via registry, disables firewall, terminates AV application including watchdog

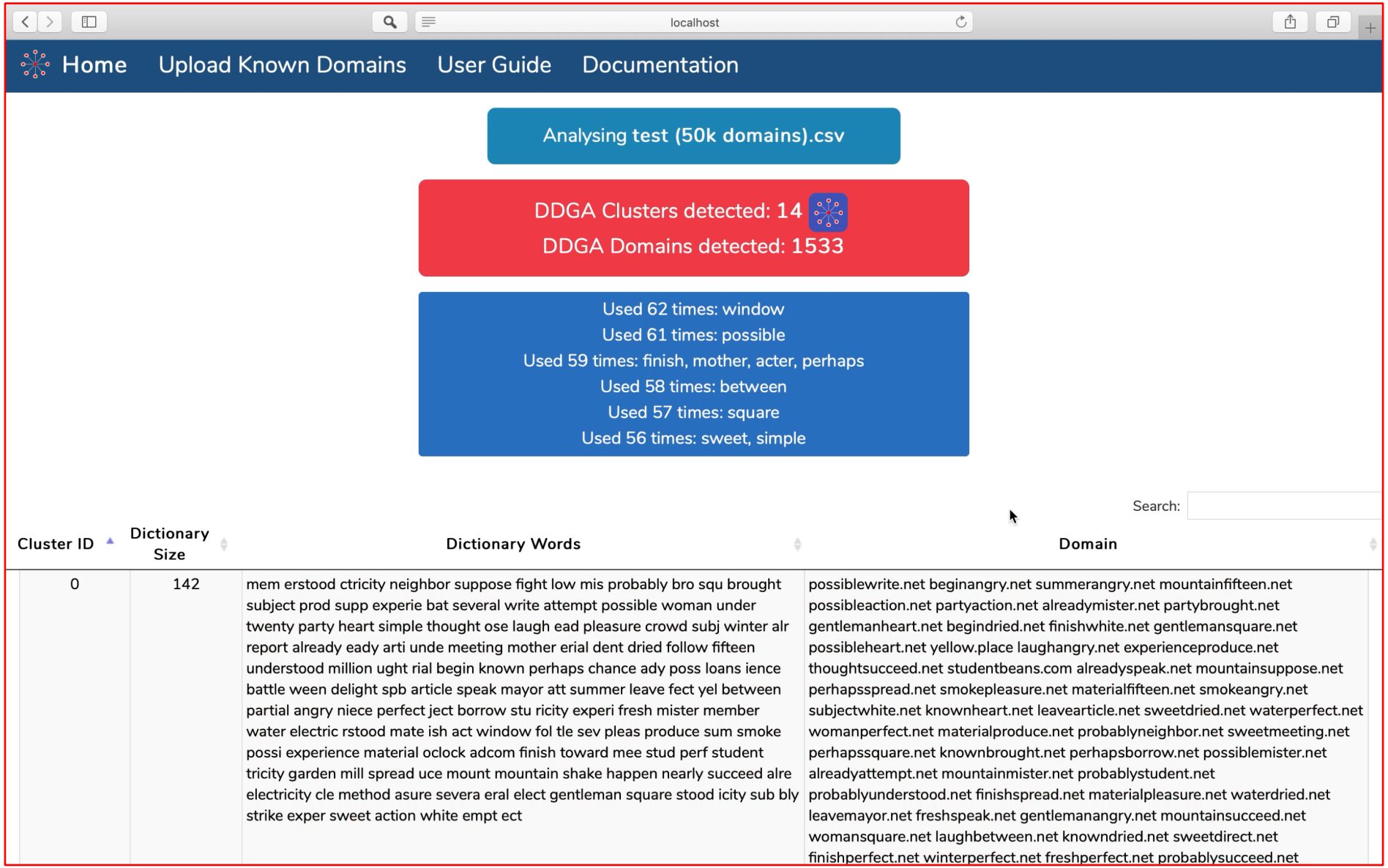
- Ran the solution against **33 million .net domains** (courtesy of **SingCert**)
- Detected **hundreds of clusters**
- Verified results in **VirusTotal**
- Domains identified in the cluster are domains used in **Bayrob Malware**

Application Tour

Using the Application



Analysing Results



Usages and Future Development

Threat Hunting

Monitoring Domains
at the National Level

Improve Clustering Techniques
Improve Classification Accuracy

Thank You

Thank You

CSA SINGAPORE

SingCERT
Singapore Computer Emergency Response Team