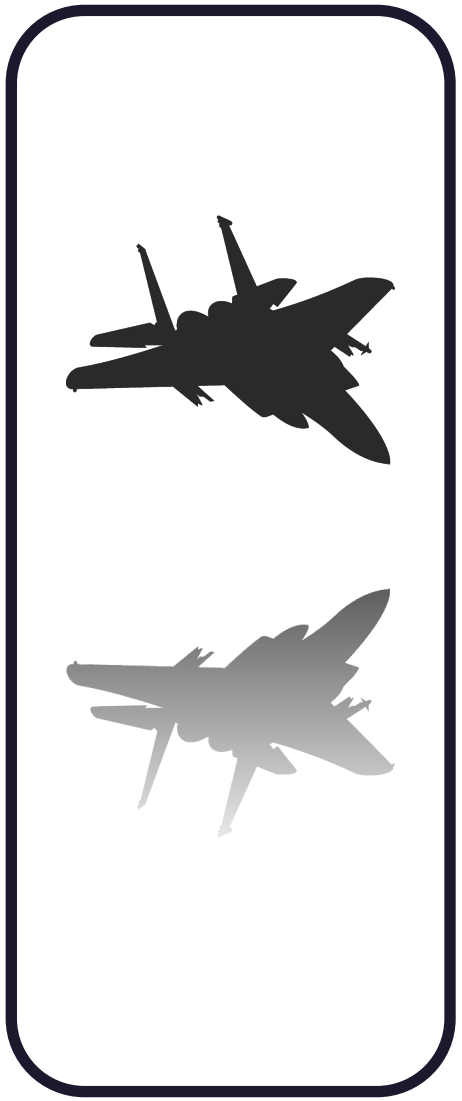
Enfilade: A Tool to Detect Infections in MongoDB Instances



Aditya K Sood and Rohit Bansal

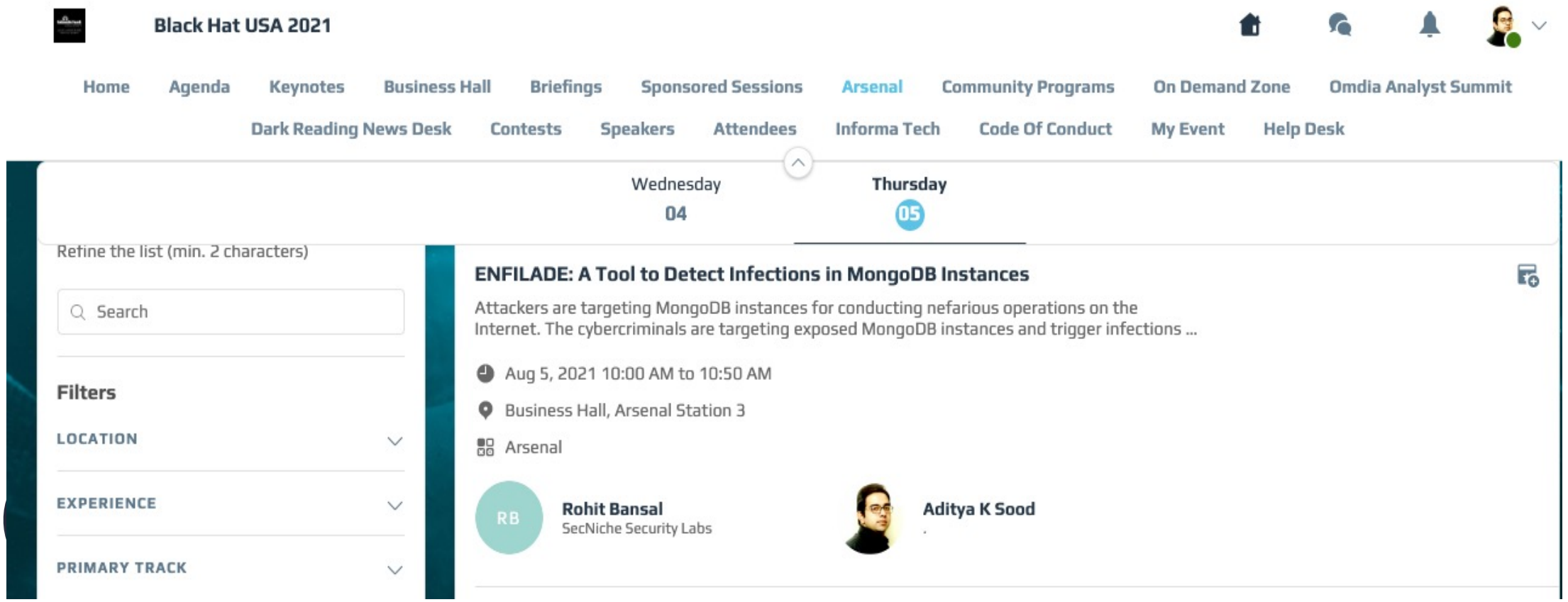
BlackHat USA Arsenal 2021

Disclaimer

Tool presented in this talk is for sharing research with security community to strengthen the intelligence efforts for enhancing the security of critical services on the internet.

Note: Due to Covid-19 Delta variant challenge and travel restrictions, we were not able to present in-person at the BlackHat USA Arsenal 2021.

BlackHat Arsenal USA 2021



Enfilade Project Team

- Dr. Aditya K Sood
  - Security Practitioner and Researcher
  - Working in the security field for more than 13 years
  - Regular speaker at industry leading security conferences
  - Author of “Targeted Cyber Attacks” and “Empirical Cloud Security” Books
  - W: https://www.adityaksood.com
  - T: @adityaksood
  - LinkedIn: https://www.linkedin.com/adityaks

MongoDB Threats Research and Tool Development

- Rohit Bansal
  - Principal Researcher, SecNiche Security Labs
  - https://secniche.org/

MongoDB Threats Research. And Intelligence Collection

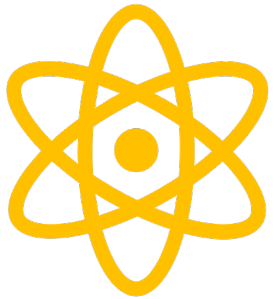
Mongodb threats and attacks landscape

MongoDB is subject to continual attacks when exposed to the internet

22,900 MongoDB Databases Affected in Ransomware Attack

An attacker scanned for databases misconfigured to expose information and wiped the data, leaving a ransom note behind.

Beware Of This Internet Cat's Meow —It Destroys Databases



Ongoing Meow attack has nuked >1,000 databases without telling anyone why

Ongoing attack hitting unsecured data leaves the word "meow" as its calling card.

MongoDB Ransomware Compromises Double in a Day

Ransomware targeting MongoDB databases threatens to report victims for GDPR breach

New 'Meow' attack has deleted almost 4,000 unsecured databases

Introducing Enfilade

- Enfilade: A tool to detect potential infections in MongoDB instances
- In this version of the tool, the following modules are supported:
  - MongoDB instances information gathering and reconnaissance (inline)
  - MongoDB instances exposure on the Internet (authentication checks)
  - MongoDB instances admin privileges assessment
  - Detecting potential ransomware infections in the MongoDB instances
    - Basic
    - Intrusive
  - Detecting potential botnet infections such as meow botnet
    - Basic
    - Intrusive

Introducing Enfilade

```
$ python enfilade.py
        ----------------------------------------------------------------------


          _____  ____________   ___    ___  ____
         / __/ |/ / __/  _/ /  / _ | / _ \/ __/
        / _//    / _/_/ // /__/ __ |/ // / _/
       /___/_/|_/_/ /___/____/_/ |_/____/___/

        ENFILADE : A Tool to Detect Potential Infections in MongoDB Deployments !
        Authored by: Aditya K Sood {https://adityaksood.com}

        ----------------------------------------------------------------------

[-] Error identified in the indexing, please check the tool usage.

[-] usage: enfilade.py <mongodb host (local or remote)> <mongodb service port> <module_name>
[*] modules: [verify_auth] | [dump_info] | [admin_access_verify] | [basic_check_ransomware] | [intrusive_check_ransomware]
[*]      : verify_auth --> check if MongoDB interface is <EXPOSED>
[*]      : dump_info --> dump information of the MongoDB instance
[*]      : admin_access_verify --> check if admin commands are allowed to run [adding user <enfilade> with role <root>]
[*]      : basic_check_ransomware --> check for basic <RANSOMWARE> indicators on the remote MongoDB instance
[*]      : basic_check_meow_bot --> check for basic <MEOW BOT> indicators on the remote MongoDB instance
[*]      : intrusive_check_ransomware --> conduct detailed analysis for <RANSOMWARE> indicators/infections on the remote MongoDB instance
[*]      : intrusive_check_meow_bot --> conduct detailed analysis for <MEOW BOT> indicators/infections on the remote MongoDB instance

[*] example: enfilade.py 127.0.0.1 27017 ransomware
```

Detecting unauthenticated mongodb instances

```
$ python enfilade.py 1[redacted]7 27017 verify_auth

[#] Checking the <GEOIP> status of the MongoDB instance ......
[*] MongoDB instance is located in <US> | <America/New_York>

[*] MongoDB instance identifier is constructed as: mongodb://1[redacted]:27017

[*] Validating authentication: checking if MongoDB interface is open to access..

[*] Target : <1[redacted]:27017>
[*] Connection established, <[UNAUTHENTICATED/EXPOSED MONGODB]> instance running....


[*] Use the module <[dump_info]> to retrieve MongoDB server information.

[*] Request processed successfully ! exiting !
```

```
$ python enfilade.py[redacted] 27017 dump_info

[#] Checking the <GEOIP> status of the MongoDB instance ......
[*] MongoDB instance is located in <US> | <America/New_York>

[*] MongoDB instance identifier is constructed as: mongodb://1[redacted]27017

[*] Validating authentication: checking if MongoDB interface is open to access..

[*] Target : <1[redacted]27017>
[*] Connection established, <[UNAUTHENTICATED/EXPOSED MONGODB]> instance running....

[*] ----------------- <[DUMPING MONGODB SERVER INFORMATION]> ----------------------

[*] storageEngines   :       [u'devnull', u'ephemeralForTest', u'mmapv1', u'wiredTiger']
[*] maxBsonObjectSize   :       16777216
[*] ok   :       1.0
[*] bits   :       64
[*] modules   :       []
[*] openssl   :       {u'compiled': u'OpenSSL 1.0.1e-fips 11 Feb 2013', u'running': u'OpenSSL 1.0.0-fips 29 Mar 2010'}
[*] javascriptEngine   :       mozjs
[*] version   :       3.6.13
[*] gitVersion   :       db3c76679b7a3d9b443a0e1b3e45ed02b88c539f
[*] versionArray   :       [3, 6, 13, 0]
[*] debug   :       False
[*] buildEnvironment   :       {u'cxxflags': u'-Woverloaded-virtual -Wno-maybe-uninitialized -std=c++14', u'cc': u'/opt/mon
godbtoolchain/v2/bin/gcc: gcc (GCC) 5.4.0', u'linkflags': u'-pthread -Wl,-z,now -rdynamic -Wl,--fatal-warnings -fstack-prot
ector-strong -fuse-ld=gold -Wl,--build-id -Wl,--hash-style=gnu -Wl,-z,noexecstack -Wl,--warn-execstack -Wl,-z,relro', u'dis
tarch': u'x86_64', u'cxx': u'/opt/mongodbtoolchain/v2/bin/g++: g++ (GCC) 5.4.0', u'ccflags': u'-fno-omit-frame-pointer -fno
-strict-aliasing -ggdb -pthread -Wall -Wsign-compare -Wno-unknown-pragmas -Winvalid-pch -Werror -O2 -Wno-unused-local-typed
efs -Wno-unused-function -Wno-deprecated-declarations -Wno-unused-but-set-variable -Wno-missing-braces -fstack-protector-st
rong -fno-builtin-memcmp', u'target_arch': u'x86_64', u'distmod': u'amazon', u'target_os': u'linux'}
[*] sysInfo   :       deprecated
[*] allocator   :       tcmalloc
[*] ----------------------------------------------------------------------------


[*] Request processed successfully ! exiting !
```

Privilege abuse: admin access (user creation)

```
$ python enfilade.py [redacted]09 27017 admin_access_verify

[#] Checking the <GEOIP> status of the MongoDB instance ......
[*] MongoDB instance is located in <TW> | <Asia/Taipei>

[*] MongoDB instance identifier is constructed as: mongodb://[redacted]09:27017

[*] Validating admin access: checking if MongoDB allows execution for admin commands....

[*] Target : <[redacted]09:27017>
[*] Connection established, trying to add user <enfilade> to the <admin> database in the target MongoDB instance ...
.

[*] Checking if the <admin> database exists on MongoDB instance: <[redacted]109:27017]>
[-] Admin database doesn't exist: configured database: <[READ__ME_TO_RECOVER_YOUR_DATA]>, not initiating the command
 to add user <[enfildae], try manually...
[*] Admin database exists on the MongoDB instance.

[*] ------------------------------------------------------------------------------
[*] Trying to add user <enfilade> with password <enfilade> to the MongoDB instance.
[*] ------------------------------------------------------------------------------


[*] Verifying whether the user <enfilade> has been added or not...enumerating the <[users]>

[*] <[SUCCESS]> user <enfilade> has been successfully added to the MongoDB instance: [[redacted]09:27017]>
{u'ok': 1.0, u'users': [{u'mechanisms': [u'SCRAM-SHA-1', u'SCRAM-SHA-256'], u'_id': u'admin.enfilade', u'db': u'admi
n', u'user': u'enfilade', u'roles': [{u'db': u'admin', u'role': u'root'}]}]}

[*] ------------------------------------------------------------------------
[*] Potential high privileges command can be executed on the MongoDB instance.
[*] ------------------------------------------------------------------------

[*] Request processed successfully ! exiting !
```

Detecting Ransomware Infections (Basic)

```
$ python enfilade.py 1[redacted]137 27017 basic_check_ransomware

[#] Checking the <GEOIP> status of the MongoDB instance ......
[*] MongoDB instance is located in <US> | <America/New_York>

[*] MongoDB instance identifier is constructed as: mongodb://1[redacted]6.137:27017

[*] Target : <[redacted].137:27017>
[*] Initiating <[BASIC CHECKS]> for <[RANSOMWARE DETECTION LOGIC]>....


[*] Checking for potential traces of ransomware......

[*] Database with potential ransom trace detected......
[D] Suspicious database detected: <[READ__ME_TO_RECOVER_YOUR_DATA]>


[*] Use the module <[intrusive_check_ransomware]> for aggresive analysis.

[*] Request processed successfully ! exiting !
```

Detecting Ransomware Infections (Intrusive)

```
$ python enfilade.py [redacted]137 27017 intrusive_check_ransomware

[#] Checking the <GEOIP> status of the MongoDB instance ......
[*] MongoDB instance is located in <US> | <America/New_York>

[*] MongoDB instance identifier is constructed as: mongodb://[redacted]137:27017

[*] Target : <[redacted]137:27017>
[*] Initiating <[INTRUSIVE CHECKS]> for <[RANSOMWARE DETECTION LOGIC]>....

[*] Dumping the identifiers of all the databases on: <[[redacted]137:27017]>
[D] READ__ME_TO_RECOVER_YOUR_DATA
[D] admin
[D] config

[*] Checking for potential traces of ransomware notifications and messages......

[*] Database with potential ransom trace detected......
[D] Suspicious database detected: <[READ__ME_TO_RECOVER_YOUR_DATA]>

[C] Suspicious collection name with ransomware trace detected...... <[README]>
[C] Suspicious collection handle: Collection(Database(MongoClient(host=['18.221.206.137:27017'], document_class=dict, tz_aware=False, connect=True,
 serverselectiontimeoutms=5000), u'READ__ME_TO_RECOVER_YOUR_DATA'), u'README')

[*] Dumping the suspicious collection records for potential <[RANSOMWARE]> messages and notifications
[*] {u'content': u'All your data is a backed up. You must pay 0.03 BTC to 15EyXBgZi88pqyN9dapDpqhX5kfsnMiWLK 48 hours for recover it. After 48 hour
s expiration we will leaked and exposed all your data. In case of refusal to pay, we will contact the General Data Protection Regulation, GDPR and
notify them that you store user data in an open form and is not safe. Under the rules of the law, you face a heavy fine or arrest and your base dum
p will be dropped from our server! You can buy bitcoin here, does not take much time to buy https://localbitcoins.com or https://buy.moonpay.io/ Af
ter paying write to me in the mail with your DB IP: myDBxm3@recoverme.one and you will receive a link to download your database dump.', u'_id': Obj
ectId('60e70d949eb05c6549782eff')}

[*] Target <[redacted].137:27017]> is potentially infected with <[RANSOMWARE]>

[*] Request processed successfully ! exiting !
```

Detecting MEOW Botnet Infections (Basic)

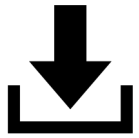
```
$ python enfilade.py [redacted]27017 basic_check_meow_bot

[#] Checking the <GEOIP> status of the MongoDB instance ......
[*] MongoDB instance is located in <CN> | <Asia/Shanghai>

[*] MongoDB instance identifier is constructed as: mongodb:/[redacted]:27017

[*] Target : <[redacted]:27017>
[*] Initiating <[BASIC CHECKS]> for <[MEOW BOT DETECTION LOGIC]>....


[*] Checking for potential traces of meow bot ......
[D] Suspicious database detected with <[meow bot]> infection: <[kf1u3g846a-meow]>
{u'storageSize': 0, u'ok': 1.0, u'avgObjSize': 0, u'db': u'item', u'indexes': 0, u'objects': 0, u'collections': 0, u'fileSize': 0,
 u'numExtents': 0, u'dataSize': 0, u'indexSize': 0}


[D] Suspicious database detected with <[meow bot]> infection: <[o95y3vzpcu-meow]>
{u'storageSize': 0, u'ok': 1.0, u'avgObjSize': 0, u'db': u'item', u'indexes': 0, u'objects': 0, u'collections': 0, u'fileSize': 0,
 u'numExtents': 0, u'dataSize': 0, u'indexSize': 0}


[D] Suspicious database detected with <[meow bot]> infection: <[6n9zo13is9-meow]>
{u'storageSize': 0, u'ok': 1.0, u'avgObjSize': 0, u'db': u'item', u'indexes': 0, u'objects': 0, u'collections': 0, u'fileSize': 0,
 u'numExtents': 0, u'dataSize': 0, u'indexSize': 0}


[D] Suspicious database detected with <[meow bot]> infection: <[mctu2uq8yu-meow]>
{u'storageSize': 0, u'ok': 1.0, u'avgObjSize': 0, u'db': u'item', u'indexes': 0, u'objects': 0, u'collections': 0, u'fileSize': 0,
 u'numExtents': 0, u'dataSize': 0, u'indexSize': 0}


[*] [Suggestion] Use the module <[intrusive_check_meow_bot]> for aggresive analysis.

[*] Request processed successfully ! exiting !
```

Detecting MEOW Botnet Infections (Intrusive)

```
$ python enfilade.py [redacted]27017 intrusive_check_meow_bot

[#] Checking the <GEOIP> status of the MongoDB instance ......
[*] MongoDB instance is located in <CN> | <Asia/Shanghai>

[*] MongoDB instance identifier is constructed as: mongodb://[redacted]:27017

[*] Target : <[redacted]27017>
[*] Initiating <[BASIC CHECKS]> for <[MEOW BOT DETECTION LOGIC]>....


[*] Checking for potential traces of meow bot ......
[C] Suspicious collection name with <meow_bot> trace detected...... <[meow]>
[C] Suspicious collection handle: Collection(Database(MongoClient(host=[[redacted]:27017'], document_class=dict, tz_aware=False,
connect=True, serverselectiontimeoutms=5000), u'kf1u3g846a-meow'), u'meow')

[*] Dumping the suspicious collection records for potential <[MEOW BOT]> messages and notifications
[*] {u'_id': ObjectId('5f2331a4f50c0c67a8adc915')}

[*] Target <[[redacted]:27017]> is potentially infected with <[MEOW BOT]>
[C] Suspicious collection name with <meow_bot> trace detected...... <[system.indexes]>
[C] Suspicious collection handle: Collection(Database(MongoClient(host=[[redacted]:27017'], document_class=dict, tz_aware=False,
connect=True, serverselectiontimeoutms=5000), u'kf1u3g846a-meow'), u'system.indexes')

[*] Dumping the suspicious collection records for potential <[MEOW BOT]> messages and notifications
[*] {u'ns': u'kf1u3g846a-meow.meow', u'name': u'_id_', u'key': {u'_id': 1}, u'v': 1}

[*] Target <[[redacted]:27017]> is potentially infected with <[MEOW BOT]>
[C] Suspicious collection name with <meow_bot> trace detected...... <[meow]>
[C] Suspicious collection handle: Collection(Database(MongoClient(host=[[redacted]:27017'], document_class=dict, tz_aware=False,
connect=True, serverselectiontimeoutms=5000), u'o95y3vzpcu-meow'), u'meow')

[*] Target <[[redacted]:27017]> is potentially infected with <[MEOW BOT]>

[*] Request processed successfully ! exiting !
```

Enfilade Tool: Demo

Enfilade Tool: Download



https://github.com/adityaks/enfilade

Questions and Queries