SLIDE 1

Logging [redacted]

92.63.107.227 - - [04/Nov/2020:06:30:48 +0000] "GET /ru/hosted-open-vpn-server/ HTTP/1.1" 301 169 "-" "python-requests/2.11.1" "-" 92.63.107.227 - - [04/Nov/2020:06:30:49 +0000] "GET /ru/data-engineering-course/ HTTP/1.1" 301 169 "-" "python-requests/2.11.1" "-" 213.180.203.50 - - [04/Nov/2020:06:36:07 +0000] "GET / HTTP/1.1" 301 169 "-" "Mozilla/5.0 (compatible; YandexMetrika/2.0; +http://yandex.com/bots yabs01)" "-" 114.119.160.75 - - [04/Nov/2020:06:36:41 +0000] "GET /robots.txt HTTP/1.1" 301 169 "-" "(compatible;PetalBot;+https://aspiegel.com/petalbot)" "10.179.80.67" 90.180.35.207 - - [04/Nov/2020:06:47:11 +0000] "GET / HTTP/1.0" 301 169 "-" "-" "-" 46.246.122.77 - - [04/Nov/2020:06:53:22 +0000] "GET / HTTP/1.1" 301 169 "" "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.3359.181 Safari/537.36" "-" 66.249.76.16 - - [04/Nov/2020:06:53:30 +0000] "GET / HTTP/1.1" 301 169 "-" "Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)" "-"

SLIDE 2

[redacted] [redacted] [redacted] ([redacted])?

[redacted] — [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] ([redacted]).

[redacted] [redacted] [redacted] [

SLIDE 3

[redacted]?

[redacted] ([redacted]) [redacted] [redacted] — [redacted] [redacted].

[redacted] [redacted] [redacted],

[redacted] [redacted] [redacted],

[redacted] [redacted] [redacted],

[redacted] [redacted]

[redacted].

[redacted] [redacted] [redacted]:

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted],

[redacted] [redacted],

[redacted] [redacted] / [redacted] API [redacted] [redacted] [redacted],

[redacted] [redacted] [redacted] ([redacted] [redacted] [redacted] [redacted] [redacted]-[redacted]),

[redacted] [redacted] email [redacted]

[redacted] [redacted] ([redacted] [redacted] [redacted])

[redacted] [redacted] [redacted] [redacted]: [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] SMS — [redacted] [redacted] [redacted] [redacted] [redacted].

SLIDE 4

[redacted]?

[redacted]:

[redacted]

[redacted], [redacted]:

```
2023-08-12 17:49:37 User 'johndoe' successfully logged in
```

[redacted].

[redacted] 3 [redacted]:

[redacted]?

[redacted]

[redacted]?

«[redacted]» — [redacted], [redacted], [redacted], [redacted], [redacted], [redacted].

[redacted], [redacted].

SLIDE 5

[redacted]: [redacted]

[redacted].

«Step initiate» «OK Transaction failed»

[redacted].

«User 'johndoe' successfully logged in from IP 1.2.3.4» «Transaction 1287786 reverted due to lost connection to DB»

[redacted].

SLIDE 6

[redacted]: [redacted] / [redacted]

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted], [redacted]:

«INFO Checking password hash for 'johndoe'» «DEBUG Hashes don't match» «ERROR Operation failed»

[redacted], [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted]: [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted].

SLIDE 7

[redacted]: [redacted]

[redacted] — [redacted]. As simple as that. [redacted] ([redacted] ASCII-[redacted]) [redacted], [redacted] [redacted].

[redacted] — [redacted], [redacted], [redacted]. [redacted] [redacted], [redacted] [redacted] [redacted].

[redacted], [redacted]:

Call func DB OK

[redacted], [redacted]:

Connection to database successfully established

SLIDE 8

[redacted]: [redacted]

[redacted], [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted],
[redacted] ISO8601:

2022-10-02 11:41:42,612

[redacted],

[redacted],

[redacted],

[redacted] ([redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]).

[redacted] [redacted] — [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted], [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] 10 000 [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

SLIDE 9

[redacted]: [redacted]

[redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] — [redacted] [redacted] [redacted], [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted].

[redacted]:

[redacted] [redacted] [redacted] [redacted]-[redacted] [redacted] [redacted] IP-[redacted], [redacted] [redacted] [redacted] 2023-09-14 14:22:45,329 172.0.0.1 GET /sitemap.xml 200

[redacted] [redacted] [redacted]-[redacted] — [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] HTTP-[redacted] [redacted] 2023-03-10 16:59:23,556 Request to https://boogle.com failed with 403

[redacted] [redacted] [redacted] [redacted] — id [redacted], [redacted] [redacted] [redacted] 2023-02-19 00:19:18,831 Transaction 309841 successful: $100500 transferred from 940384 to 473923

[redacted] [redacted] [redacted]-[redacted] — telegram_id [redacted] [redacted] [redacted] [redacted] ([redacted] / handler), [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]

SLIDE 10

[redacted]: [redacted]: [redacted]

[redacted], [redacted] [redacted], [redacted] [redacted] [redacted] / [redacted], [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

2023-05-15 05:59:46,102 Division by zero at mymodule.divide_by(0):18

[redacted] [redacted] [redacted] — [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]-[redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]

SLIDE 11

[redacted]: [redacted]

[redacted]. [redacted] — [redacted], [redacted] — [redacted].

[redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted]:

[redacted] ([redacted]) [redacted] [redacted] [redacted] [redacted],

[redacted],

[redacted]:

2023-02-19 00:19:18 Transaction 309841 successful: $100500 transferred from 940384 to 473923

[redacted], [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted]., [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted]:

2023-02-19 00:19:18 Transaction successful. Transaction_id: 309841, amount: 100500, From: 940384, To: 473923

SLIDE 12

[redacted]: [redacted] / [redacted]

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] JSON:

{"timestamp":1676755158831,"code":"OK","transaction":{"id":309841,"amount":100500,"from":940384,"to":473923}}

[redacted] — [redacted] [redacted], [redacted] [redacted] Ctrl+F [redacted] [redacted], grep [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] / [redacted] [redacted]. [redacted], [redacted] [redacted] [redacted]-[redacted] [redacted] user_id, [redacted] [redacted]-[redacted] client identificator, [redacted]-[redacted] login, [redacted] [redacted]-[redacted] authorization, [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted], [redacted]/[redacted] [redacted] [redacted] [redacted] [redacted].

SLIDE 13

[redacted]: [redacted]

[redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted]. [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted]:

[redacted], [redacted] [redacted], [redacted] [redacted];

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted];

[redacted], [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted].

SLIDE 14

[redacted], [redacted] [redacted] [redacted] [redacted]? [redacted] [redacted]

[TABLE CONTENT]

```
Numerical Code | Severity
0 | Emergency: system is unusable
1 | Alert: action must be taken immediately
2 | Critical: critical conditions
3 | Error: error conditions
4 | Warning: warning conditions
5 | Notice: normal but significant condition
6 | Informational: informational messages
7 | Debug: debug-level messages
```

[redacted] «[redacted] [redacted] [redacted], [redacted] [redacted] [redacted]» — [redacted] [redacted], [redacted] [redacted] [redacted] (severity levels).

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] — [redacted]-[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted]-[redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] 1980 [redacted] [redacted] [redacted] syslog:

SLIDE 15

[redacted]

[redacted], [redacted]. [redacted], [redacted] Python [redacted] ([redacted]. [redacted]).

[redacted], [redacted], [redacted], [redacted], [redacted].

[redacted], [redacted]. [redacted], [redacted] DEBUG, [redacted], [redacted], [redacted] production-[redacted], [redacted]-[redacted]. [redacted], [redacted], [redacted], [redacted].

[redacted], [redacted]-[redacted]. [redacted], [redacted] INFO [redacted], [redacted]-[redacted] [redacted]-[redacted]. [redacted] ERROR [redacted] CRITICAL [redacted], [redacted], [redacted].

[TABLE CONTENT]

[redacted]	Code	[redacted]
DEBUG	10	[redacted], [redacted].
INFO	20	[redacted], [redacted], [redacted], [redacted].
WARNING	30	[redacted], [redacted]-[redacted], [redacted]-[redacted], [redacted].
ERROR	40	[redacted]-[redacted]-[redacted], [redacted].
CRITICAL	50	[redacted], [redacted]-[redacted].

[SLIDE NOTES]

DEBUG — [redacted], [redacted].

INFO — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]/[redacted].

WARNING — [redacted] [redacted]. [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

ERROR — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted].

CRITICAL — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

SLIDE 16

[redacted]

[redacted] WARNING, ERROR, CRITICAL [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted].

[redacted] [redacted] DEBUG [redacted] INFO [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] — [redacted]: ✓ [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] (**[redacted] [redacted] [redacted] [redacted]**); ✓ [redacted] [redacted] [redacted]-[redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted], [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted].

[redacted] [redacted] [redacted] (**[redacted]**, [redacted] [redacted]) [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] (**[redacted]** [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]), [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] (**[redacted]** [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]).

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] — [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] — [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]-[redacted].

SLIDE 17

[redacted]: [redacted]

[redacted], [redacted]. [redacted]. [redacted] print(). [redacted], [redacted], [redacted]-[redacted], [redacted]. [redacted] — [redacted].

[redacted], [redacted], [redacted], [redacted]. [redacted]. [redacted]: rsyslogd, logrotate, Sentry, ELK.

SLIDE 18

[redacted]

[redacted]:

[redacted]:

[redacted],

[redacted],

[redacted].

[redacted]:

[redacted],

[redacted].

[redacted]:

[redacted],

[redacted]-[redacted].

[redacted]:

[redacted],

[redacted],

[redacted],

[redacted],

[redacted].

[redacted] — [redacted], [redacted] — [redacted], [redacted] [redacted], [redacted], [redacted].

SLIDE 19

[redacted]

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]-[redacted] [redacted] [redacted] [redacted]. [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted], [redacted] [redacted] error.log. [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] 24 [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] 24 [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] — [redacted] 0 [redacted]. [redacted] [redacted] [redacted] — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] errors.log.1, [redacted], [redacted] [redacted] [redacted], [redacted] errors.log.2 [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] 24, [redacted] [redacted] 48 [redacted].

$ ls /var/logs/ keybagd.log.0 keybagd.log.2 usermanagerd.log.0 keybagd.log.1 keybagd.log.3 usermanagerd.log.1

[redacted] [redacted] — [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] — logrotate. [redacted] [redacted] [redacted] [redacted] [redacted]: [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] / [redacted] [redacted] [redacted], [redacted] [redacted] [redacted].

SLIDE 20

[redacted]: logrotate

/etc/logrotate.d/myapp:

/var/log/myapp/error.log { rotate 7 # [redacted] [redacted] 7 [redacted] [redacted] daily # [redacted] [redacted] compress # [redacted] [redacted] [redacted] notifempty # [redacted] [redacted], [redacted] [redacted] [redacted] create 660 mebossuser mebossuser # [redacted] / user [redacted] [redacted] [redacted] }

[redacted] [redacted], [redacted], [redacted] [redacted], [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted] rsync.

SLIDE 21

[redacted] Python / [redacted] logging

[redacted] Python — [redacted] print [redacted] [redacted], [redacted] open. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] Python [redacted] logging. [redacted], [redacted] [redacted] debug(), info(), warning(), error(), critical(), [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] logging [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] sys.stderr ([redacted] [redacted], [redacted] [redacted]), [redacted] [redacted] [redacted] [redacted] WARNING. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] WARNING (DEBUG [redacted] INFO) [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

import logging logging.info('For your information') # [redacted], [redacted]. [redacted]. INFO < WARNING logging.warning('I warn you!') # [redacted] [redacted] [redacted] [redacted]

[redacted] [redacted] [redacted] [redacted]:

WARNING:root:I warn you!

[redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] INFO [redacted] [redacted] WARNING.

SLIDE 22

[redacted]

[redacted]:

import logging logging.basicConfig(filename='example.log', level=logging.DEBUG) logging.debug("Let's see if it works") logging.info('For your information') logging.warning('I warn you!') logging.error('Something bad happened pls help!')

[redacted] filename, [redacted] level [redacted]. [redacted] DEBUG, [redacted], [redacted] [redacted].

basicConfig() [redacted], [redacted] debug(), info() [redacted]. [redacted] basicConfig() [redacted].

SLIDE 23

[redacted] logging

[redacted].

[redacted] logging — [redacted]:

[redacted] (loggers) — [redacted];

[redacted] (handlers) — [redacted];

[redacted] (filters) — [redacted];

[redacted] (formatters) — [redacted].

[redacted] LogRecord, [redacted].

SLIDE 24

Logging Flow

SLIDE 25

[redacted]

[redacted], [redacted] [redacted] [redacted] [redacted] [redacted]:

import logging counter = '3rd' logging.warning('I am warning you for the %s time!', counter)

→ WARNING:root:I am warning you for the 3rd time!

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted]:

logging.basicConfig(format='%(asctime)s %(levelname)s %(message)s') logging.warning('Right about time!')

→ 2023-08-21 13:44:38,417 WARNING Right about time!

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] levelname [redacted] [redacted] [redacted] message, [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] asctime. [redacted] [redacted]-[redacted] [redacted] [redacted] [redacted] basicConfig() ([redacted] [redacted] [redacted] [redacted] [redacted]).

logging.basicConfig(format='%(asctime)s %(message)s', datefmt='%d.%m.%Y %H:%M:%S') logging.warning('is when this event was logged.')

→ 21.08.2023 13:54:51 is when this event was logged.

SLIDE 26

[redacted]

[redacted] logging.basicConfig() [redacted] logging [redacted] Python [redacted], [redacted] [redacted] [redacted] format [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] LogRecord attributes.

[redacted] [redacted] [redacted] [redacted] logging.basicConfig(), [redacted] logging [redacted] [redacted] [redacted] [redacted] [redacted]:

[TABLE CONTENT]

[redacted] | [redacted]

%(asctime)s | [redacted]

%(levelname)s | [redacted]

%(levelno)s | [redacted]

%(message)s | [redacted]

%(name)s | [redacted]

%(filename)s | [redacted]

%(module)s | [redacted]

%(lineno)d | [redacted]

%(funcName)s | [redacted]

%(process)d | ID [redacted]

%(processName)s | [redacted]

```
formatter = logging.Formatter( '%(asctime)s | %(levelname)s | %(message)s' ) logger =
logging.getLogger('my_logger') handler = logging.StreamHandler()
handler.setFormatter(formatter) logger.addHandler(handler)
```

SLIDE 27

[redacted] (loggers)

[redacted] Logger [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] Logger [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] «.» [redacted] [redacted]. [redacted], file [redacted] [redacted] [redacted] file.error, file.access. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] root ([redacted] [redacted] [redacted] root, [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]).

[redacted] [redacted] [redacted] [redacted] — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]:

logger = logging.getLogger(__name__)

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] root [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted].

SLIDE 28

[redacted] / Logger

[redacted]:

Logger.setLevel() — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted]; [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] root, [redacted] [redacted] [redacted] [redacted] [redacted] ([redacted] [redacted] WARNING).

Logger.addHandler(), Logger.removeHandler(), Logger.addFilter(), Logger.removeFilter() — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] / [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted]:

Logger.debug(), Logger.info(), Logger.warning(), Logger.error(), Logger.critical() — [redacted] [redacted] [redacted] [redacted] [redacted].

Logger.log() — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted], [redacted] [redacted], [redacted] [redacted] [redacted] [redacted].

Logger.exception() — [redacted] [redacted] [redacted] Logger.error(), [redacted] [redacted] stack trace. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

logging.getLogger(loggername) — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] root, [redacted] [redacted] [redacted] [redacted].

SLIDE 29

[redacted] (handlers)

[redacted]. [redacted], FileHandler [redacted] [redacted] [redacted], SMTPHandler [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] Handler. [redacted] [redacted] Handler [redacted].

[redacted]:

setLevel() — [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] ([redacted] [redacted] [redacted], [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted]);

setFormatter() — [redacted];

addFilter(), removeFilter() — [redacted] [redacted] [redacted].

[TABLE CONTENT]

[redacted] | [redacted]

StreamHandler | [redacted] [redacted] [redacted], [redacted] [redacted] sys.stdout [redacted] sys.stderr ([redacted], [redacted] [redacted] [redacted])

FileHandler | [redacted] [redacted] [redacted]

RotatingFile- Handler | [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted]

TimedRotating-FileHandler | [redacted] [redacted] [redacted] [redacted] [redacted] ([redacted], [redacted], [redacted])

SMTPHandler | [redacted] [redacted] [redacted] email

SysLogHandler | [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] (syslog)

SLIDE 30

[redacted] (formatters)

[redacted].

logging.Formatter(fmt=None, datefmt=None, style='%')

[redacted]:

[redacted] fmt;

[redacted] datefmt;

[redacted] style [redacted] LogRecord:

'%' — [redacted] %-[redacted] ([redacted]),

'{' — [redacted] str.format() [redacted]

'$' — [redacted] string.Templates.

[redacted] strfmt = '[%(asctime)s] [%(name)s] [%(levelname)s] > %(message)s' # [redacted] datefmt = '%Y-%m-%d %H:%M:%S' # [redacted] formatter = logging.Formatter(fmt=strfmt, datefmt=datefmt)

→[2020-06-08 07:42:59] [logger] [DEBUG] > debug message →[2020-06-08 07:42:59] [logger] [INFO] > info message

SLIDE 31

[redacted]

[redacted]:

[redacted], [redacted] [redacted] [redacted] [redacted] [redacted] Python, [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] fileConfig().

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] dictConfig().

SLIDE 32

[redacted] (**[redacted]**)

import logging # [redacted] logger = logging.getLogger('exampleLogger') logger.setLevel(logging.INFO) # [redacted] console_handler = logging.StreamHandler() console_handler.setLevel(logging.INFO) # [redacted] formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s') # [redacted] console_handler.setFormatter(formatter) # [redacted] logger.addHandler(console_handler) # [redacted] ([redacted], [redacted] DEBUG) logger.debug('debug message') logger.info('info message') logger.warning('warn message') logger.error('error message') logger.critical('critical message')

SLIDE 33

[redacted]

[redacted] fileConfig() [redacted] [redacted] dictConfig().

[redacted] Python.

import logging import logging.config # [redacted] logging.config.fileConfig('logging.conf') # [redacted] logger = logging.getLogger('exampleLogger') # [redacted] logger.debug('debug message') logger.info('info message') logger.warning('warn message') logger.error('error message') logger.critical('critical message')

SLIDE 34

logging.conf

[loggers] keys=root,exampleLogger [handlers] keys=consoleHandler [formatters] keys=simpleFormatter [logger_root] level=DEBUG handlers=consoleHandler [logger_simpleExample] level=DEBUG handlers=consoleHandler qualname=simpleExample propagate=0 [handler_consoleHandler] class=StreamHandler level=DEBUG formatter=simpleFormatter args=(sys.stdout,) [formatter_simpleFormatter] format=%(asctime)s - %(name)s - %(levelname)s - %(message)s

SLIDE 35

logging.config.dictConfig

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted]:

[redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] (JSON [redacted] YAML)

[redacted] [redacted] [redacted],

[redacted] [redacted] [redacted] [redacted] [redacted]

[redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] YAML:

handlers: console: class : logging.StreamHandler formatter: brief level : INFO filters: [allow_foo] stream : ext://sys.stdout file: class : logging.handlers.RotatingFileHandler formatter: precise filename: logconfig.log maxBytes: 1024 backupCount: 3

SLIDE 36

■■■■■■■ JSON-■■■■■■■■ ■■■■■■■■■■■■■ ■ ■■■■■■■■■■■■■■■■ ■■■■■■■■■■■

```
{
"version": 1,
"disable_existing_loggers": false,
"formatters": {
"simple": {
"format": "%(asctime)s - %(name)s - %(levelname)s - %(message)s"
},
"extra": {
"format": "%(asctime)s %(name)s %(filename)s %(lineno)s %(funcName)s %(levelname)s
%(message)s"
}
},
"handlers": {
"console_handler": {
"class": "logging.StreamHandler",
"level": "DEBUG",
"formatter": "simple",
"stream": "ext://sys.stdout"
},
"info_file_handler": {
"class": "logging.FileHandler",
"level": "INFO",
"formatter": "simple",
"filename": "info.log",
"encoding": "utf8"
},
"error_file_handler": {
"class": "logging.FileHandler",
```

"level": "ERROR",
"formatter": "extra",
"filename": "errors.log",
"encoding": "utf8"
}
},
"loggers": {
"MyApp": {
"level": "WARNING",
"handlers": ["console_handler"],
"propagate": false
},
"MyApp.MyClass1": {
"level": "DEBUG",
"handlers": ["console_handler", "info_file_handler", "error_file_handler"],
"propagate": false
},
"MyApp.MyClass2": {
"level": "ERROR",
"handlers": ["error_file_handler"],
"propagate": false
}
},
"root": {
"level": "WARNING",
"handlers": ["console_handler"]
}
}

SLIDE 37

■■■ ■ ■■■■■■■■■■■■■■■ dictConfig

import json import logging, logging.config class MyAppLogger: def __init__(self): dict = json.loads(config) logging.config.dictConfig(dict) self.logger = logging.getLogger('MyApp') # ■■■■■■■■ ■■■■■■■■■■ def getLogger(self, name): """■■■■■■■■■■ ■■■■■■■■ ■ ■■■■■■■■■"""
return logging.getLogger('MyApp.' + name) class MyClass1: def __init__(self): self.logger = MyAppLogger().getLogger(self.__class__.__name__) class MyClass2: def __init__(self): self.logger = MyAppLogger().getLogger(self.__class__.__name__) def test(object): object.logger.debug('debug message') object.logger.info('info message') object.logger.warning('warn message') object.logger.error('error message') object.logger.critical('critical message') test(MyAppLogger()) test(MyClass1()) test(MyClass2())

SLIDE 38

[redacted]

[redacted]:

2023-08-21 17:09:40,562 - MyApp - WARNING - warn message

2023-08-21 17:09:40,563 - MyApp - ERROR - error message

2023-08-21 17:09:40,563 - MyApp - CRITICAL - critical message

2023-08-21 17:09:40,564 - MyApp.MyClass1 - DEBUG - debug message

2023-08-21 17:09:40,565 - MyApp.MyClass1 - INFO - info message

2023-08-21 17:09:40,565 - MyApp.MyClass1 - WARNING - warn message

2023-08-21 17:09:40,566 - MyApp.MyClass1 - ERROR - error message

2023-08-21 17:09:40,566 - MyApp.MyClass1 - CRITICAL - critical message

info.log:

2023-08-21 17:16:19,753 - MyApp.MyClass1 - INFO - info message

2023-08-21 17:16:19,756 - MyApp.MyClass1 - WARNING - warn message

2023-08-21 17:16:19,756 - MyApp.MyClass1 - ERROR - error message 2023-08-21 17:16:19,756 - MyApp.MyClass1 - CRITICAL - critical message

errors.log:

2023-08-21 17:17:18,172 MyApp.MyClass1 main.py 93 test ERROR error message

2023-08-21 17:17:18,172 MyApp.MyClass1 main.py 94 test CRITICAL critical message

2023-08-21 17:17:18,173 MyApp.MyClass2 main.py 93 test ERROR error message

2023-08-21 17:17:18,173 MyApp.MyClass2 main.py 94 test CRITICAL critical message

SLIDE 39

[redacted] [redacted] [redacted]

[redacted] MyApp ([redacted] [redacted] MyAppLogger) [redacted] [redacted]
[redacted] console_handler, [redacted] [redacted] [redacted]
[redacted]. console_handler [redacted] [redacted] [redacted] DEBUG, [redacted] [redacted]
[redacted] MyApp — WARNING, [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]
[redacted] — WARNING, ERROR [redacted] CRITICAL.

[redacted] MyApp.MyClass1 ([redacted] [redacted] MyClass1) [redacted] [redacted]
[redacted]: console_handler, info_file_handler, error_file_handler. [redacted] [redacted]
[redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted]
[redacted] DEBUG. [redacted] info.log [redacted] [redacted] INFO [redacted] [redacted] [redacted]
[redacted] [redacted] [redacted] info_file_handler, [redacted] [redacted] errors.log — ERROR [redacted] CRITICAL.

[redacted] MyApp.MyClass2 ([redacted] [redacted] MyClass2) [redacted]
[redacted] ERROR [redacted] CRITICAL [redacted] errors.log.

SLIDE 40

«[redacted]» [redacted]

[redacted] ([redacted]) / [redacted] [redacted] [redacted] [redacted] [redacted]:

[redacted]:

datetime

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] uppercase: USER_ADDED, DOMAIN_DELETED, MAIL_SENT, PASSWORD_RECOVERY_REQUEST, SMS_SEND_ERROR [redacted].

service / script name

hostname

[redacted] [redacted] [redacted], [redacted] [redacted] [redacted] / [redacted], [redacted]:

user_id / service_id — [redacted] [redacted] [redacted]-[redacted] [redacted], [redacted] [redacted] [redacted] [redacted]

[redacted] [redacted] [redacted] [redacted] [redacted] / [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted]:

DATA

```
2023-08-27 12:12:12 | USER_ADDED | mobappbackend | cl23.mycloud.ru | 23455 | 0 |
{"Name":"Ivan Ivanov","email":"ivanoff@gmail.com","regip":"1.2.3.4"}

2023-08-27 12:12:13 | MAIL_SENT | userspam.py | cl2.mycloud.ru | 11455 | 249499 |
{"email":"prtrff@gmail.com","title":"[redacted] [redacted] [redacted] [redacted] [redacted]!"}
```

[redacted]:

[redacted] [redacted] [redacted] / [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted], [redacted]. [redacted] [redacted] [redacted] [redacted] [redacted].

SLIDE 41

[redacted] «[redacted]»

import logging default_extra_fields = { 'user_id': 0, 'key_id': 0, 'ext_params': '' } class CustAdapter(logging.LoggerAdapter): def process(self, msg, kwargs): extra_fields = {} for key in ['user_id', 'key_id', 'ext_params']: extra_fields[key] = kwargs.pop(key, self.extra[key]) kwargs['extra'] = extra_fields return msg, kwargs logger = logging.getLogger('actions') logger.setLevel(logging.INFO) handler = logging.StreamHandler() # handler = logging.FileHandler('/var/log/myapp.log') formatter = logging.Formatter(fmt = '%(asctime)s\t%(message)s\t%(user_id)s\t%(key_id)s\t%(ext_params)s', datefmt = '%d.%m.%Y %H:%M:%S',) handler.setFormatter(formatter) logger.addHandler(handler) logger = CustAdapter(logger, default_extra_fields) # [redacted] [redacted] logger.info('SOME_ACTION') logger.info('ANOTHER_ACTION', key_id = 777) logger.info('ACTION1', user_id = '123', ext_params = {'param1': 123, 'param2': ['A','B',456]}) logger.info('ACTION1', user_id = '123', key_id = 77, ext_params = {'par1': 123, 'par2': ['A','B',45]})

[redacted]: [redacted]-[redacted]. [redacted] [redacted]:

user_id

key_id

[redacted] [redacted]:

ext_params

SLIDE 42

Loguru

[redacted] logging — [redacted]! Loguru — [redacted]. «Ready to use out of the box without boilerplate code.»

SLIDE 43

[redacted] loguru [redacted] [redacted] logging?

[redacted]. [redacted] [redacted] loguru [redacted], [redacted] logging.

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] — [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] crash-[redacted] [redacted] [redacted].

[redacted]. [redacted] [redacted] [redacted] [redacted].

[redacted]. Zero-cost logger [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] ([redacted] C).

[redacted] [redacted] loguru [redacted] [redacted]:

[redacted] [redacted] [redacted] [redacted] [redacted] logging, [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] loguru [redacted] [redacted] [redacted] [redacted] [redacted] logging. [redacted], [redacted] [redacted] [redacted] [redacted] Sentry [redacted] Airbrake.

[redacted] Loguru

SLIDE 44

■■■■■■■■■■■■■ Loguru

✓ Ready to use out of the box without boilerplate

from loguru import logger logger.debug("That's it, beautiful and simple logging!")

```
2023-11-13 20:29:32.531 | DEBUG  | __main__::1 - That's it, beautiful and simple
logging!
```

✓ No Handler, no Formatter, no Filter: one function to rule them all

How to add a handler? How to set up logs formatting? How to filter messages? How to set level?

logger.add(sys.stderr, format="{time} {level} {message}", filter="my_module", level="INFO")

✓ Easier file logging with rotation / retention / compression

logger.add("file_{time}.log") logger.add("file_1.log", rotation="500 MB") # Automatically rotate too big file logger.add("file_2.log", rotation="12:00") # New file is created each day at noon logger.add("file_3.log", rotation="1 week") # Once the file is too old, it's rotated logger.add("file_X.log", retention="10 days") # Cleanup after some time logger.add("file_Y.log", compression="zip") # Save some loved space

SLIDE 45

■■■■■■■■■■■■■ Loguru

✓ Modern string formatting using braces style

logger.info("If you're using Python {}, prefer {feature} of course!", 3.6, feature="f-strings")

✓ Exceptions catching within threads or main

@logger.catch def my_function(x, y, z): # An error? It's caught anyway! return 1 / (x + y + z)

✓ Pretty logging with colors

logger.add(sys.stdout, colorize=True, format="{time} {message}")

SLIDE 46

logger.add("out.log", backtrace=True, diagnose=True) def func(a, b): return a / b def nested(c): try: func(5, c) except ZeroDivisionError: logger.exception("What?!") nested(0)

■■■■■■■■■■■■■ Loguru

✓ Fully descriptive exceptions

```
2018-07-17 01:38:43 | ERROR | __main__:nested:10 What?
```

Traceback (most recent call last):

File "test.py", line 12, in

nested(0)

■

> File "test.py", line 8, in nested

func(5, c)

■ ■ 0

■

File "test.py", line 4, in func

return a / b

■ ■ 0

■ 5

ZeroDivisionError: division by zero

SLIDE 47

■■■■■■■■■■■■ Loguru

✓ Lazy evaluation of expensive functions

Sometime you would like to log verbose information without performance penalty in production, you can use the opt() method to achieve this.

logger.opt(lazy=True).debug("If sink level <= DEBUG: {x}", x=lambda: expensive_function(2**64)) # By the way, "opt()" serves many usages logger.opt(exception=True).info("Error stacktrace added to the log message (tuple accepted too)") logger.opt(colors=True).info("Per message colors") logger.opt(record=True).info("Display values from the record (eg. {record[thread]})") logger.opt(raw=True).info("Bypass sink formatting\n") logger.opt(depth=1).info("Use parent stack context (useful within wrapped functions)") logger.opt(capture=False).info("Keyword arguments not added to {dest} dict", dest="extra")

SLIDE 48

✓ Better datetime handling

The standard logging is bloated with arguments like datefmt or msecs, %(asctime)s and %(created)s, naive datetimes without timezone information, not intuitive formatting, etc. Loguru fixes it:

```
logger.add("file.log", format="{time:YYYY-MM-DD at HH:mm:ss} | {level} | {message}")
```

✓ Suitable for scripts and libraries

For scripts config = { "handlers": [{"sink": sys.stdout, "format": "{time} - {message}"}, {"sink": "file.log", "serialize": True},], "extra": {"user": "someone"} } logger.configure(**config) # For libraries, should be your library's `__name__` logger.disable("my_library") logger.info("No matter added sinks, this message is not displayed") # In your application, enable the logger in the library logger.enable("my_library") logger.info("This message however is propagated to the sinks")

SLIDE 49

✓ Entirely compatible with standard logging

handler = logging.handlers.SysLogHandler(address=('localhost', 514)) logger.add(handler)

✓ Personalizable defaults through environment variables

```
# Linux / OSX export LOGURU_FORMAT="{time} | {message}" # Windows setx
LOGURU_DEBUG_COLOR ""
```

✓ Convenient parser

pattern = r"(?P.*) - (?P[0-9]+) - (?P.*)" # Regex / named groups caster_dict = dict(time=dateutil.parser.parse, level=int) # Transform matching groups for groups in logger.parse("file.log", pattern, cast=caster_dict): print("Parsed:", groups) # {"level": 30, "message": "Log example", "time": datetime(2018, 12, 09, 11, 23, 55)}

SLIDE 50

✓ Exhaustive notifier

Loguru can easily be combined with the great notifiers library (must be installed separately) to receive an e-mail when your program fail unexpectedly or to send many other kind of notifications: Pushover, SimplePush, Slack, Gmail, Telegram, Gitter, Pushbullet, Join, Zulip, Twilio, Pagerduty, Mailgun, PopcornNotify, StatusPage.io, iCloud, VictorOps (Splunk)

import notifiers params = { "username": "you@gmail.com", "password": "abc123", "to": "dest@gmail.com" } # Send a single notification notifier = notifiers.get_notifier("gmail") notifier.notify(message="The application is running!", **params) # Be alerted on each error message from notifiers.logging import NotificationHandler handler = NotificationHandler("gmail", defaults=params) logger.add(handler, level="ERROR")

✓ 10x faster than built-in logging

Zero-cost logger.

SLIDE 51

```
import sys from loguru import logger # [redacted] [redacted] [redacted] [redacted] [redacted]
[redacted] [redacted] [redacted] [redacted] [redacted] log_format = "{time:YYYY-MM-DD HH:mm:ss.SSS}
| " + \              "{message} | {extra[user_id]} | " + \              "{extra[key_id]}
| {extra[ext_params]}" config = {     "handlers": [ {"sink": sys.stdout, "format":
log_format} ],     "extra": { "user_id": 0, "key_id": 0, "ext_params": "" } }
logger.configure(**config) # [redacted] [redacted] [redacted] logger.info('SOME_ACTION')
logger.info('ANOTHER_ACTION', key_id = 777) logger.info('ACTION1', user_id = '123',
ext_params = {'par1': 123, 'param2': ['A','B',456]}) logger.info('ACTION2', user_id =
'123', key_id = 77, ext_params = {'par1': 123, 'par2': 'Z'})
```

Loguru custom fields

```
2023-11-13 22:35:12.579 | SOME_ACTION | 0 | 0 |

2023-11-13 22:35:12.579 | ANOTHER_ACTION | 0 | 777 |

2023-11-13 22:35:12.579 | ACTION1 | 123 | 0 | {'par1': 123, 'param2': ['A', 'B', 456]}

2023-11-13 22:35:12.579 | ACTION2 | 123 | 77 | {'par1': 123, 'par2': 'Z'}
```

[redacted]: [redacted]-[redacted]. [redacted] [redacted]:

user_id

key_id

[redacted] [redacted]:

ext_params

SLIDE 52

[illegible]

[illegible]: [illegible], [illegible]:

[illegible] / [illegible] / [illegible] / [illegible] ([illegible], [illegible])

[illegible], "[illegible], [illegible] 12345 [illegible] 2023 [illegible]"

[illegible]

[illegible] ([illegible], [illegible].). [illegible] logrotate [illegible], [illegible] / [illegible]. [illegible], [illegible] — [illegible] ([illegible]).

[illegible] / [illegible]:

[illegible] ([illegible] «[illegible]», [illegible]., [illegible])

[illegible] ([illegible]) — [illegible]

[illegible]: [illegible] column-based [illegible] (clickhouse [illegible].):

[illegible]

[illegible]

[illegible]/[illegible] ([illegible])

SLIDE 53

Clickhouse + web logs + ■■■■■■■

SLIDE 54

[redacted]: ELK

ELK – [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]: Elasticsearch, Logstash [redacted] Kibana. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

E = Elasticsearch —[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] Apache Lucene.

L = Logstash — [redacted] [redacted] [redacted] [redacted]: [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

K = Kibana Kibana — [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. Kibana [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] ELK?

Logstash [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

Elasticsearch [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

Kibana [redacted] [redacted] [redacted] [redacted].

[redacted]:

[redacted] — [redacted] [redacted] [redacted] [redacted]

[redacted] [redacted] [redacted], “[redacted] [redacted]” [redacted] [redacted] [redacted] [redacted].

SLIDE 55

Graylog

Graylog — [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] Elasticsearch, MongoDB [redacted] Scala.

[redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted], [redacted] [redacted]-[redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] (**[redacted]**, [redacted] [redacted]) - [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] Graylog

[redacted] open source [redacted].

[redacted] [redacted] [redacted] [redacted] [redacted].

[redacted] [redacted], [redacted] [redacted], [redacted] [redacted].

«[redacted]» [redacted] [redacted], [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] ELK [redacted] [redacted] [redacted].

SLIDE 56

Sentry

Sentry — [redacted] (exception), [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted], [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] PHP, Node.js, Python, Ruby, C #, Java, Go, React, Angular, Vue, JavaScript [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] [redacted].

[redacted] [redacted] [redacted] BackEnd [redacted] [redacted] FrontEnd:

[redacted] [redacted] Sentry [redacted] [redacted], [redacted], [redacted] [redacted] [redacted] [redacted]. [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] JavaScript. [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted], [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted] [redacted].

```
import logging, sentry_sdk from sentry_sdk.integrations.logging import LoggingIntegration
sentry_logging = LoggingIntegration( level=logging.INFO, event_level=logging.ERROR )
sentry_sdk.init( dsn="https://exampleKey@o0.ingest.sentry.io/0", integrations=[ sentry_logging, ], )
```

SLIDE 57

[redacted]

https://github.com/xtrueman/prog_instruments/blob/main/Logging.md

logging

Logging HOWTO

Loggint Cookbook