

# 15 - Elastic Stack

Elasticsearch nemusí sloužit jen jako vyhledávací nástroj, Používá se také pro ukládání logů a metrik z různých systémů. Samotný Elasticsearch je však pouze úložiště, neobsahuje grafické uživatelské rozhraní ani nástroje pro sběr dat - k tomu slouží další samostatné nástroje.

## Logstash

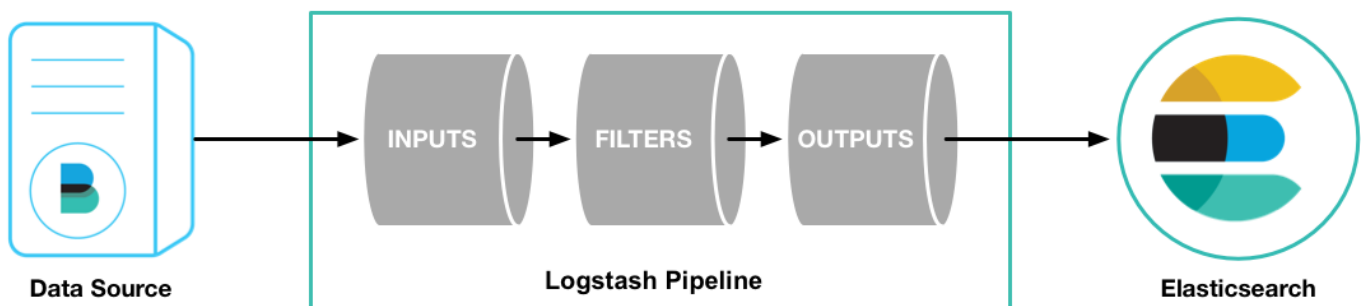
Logstash je nástroj, který běží jako samostatná aplikace a umožňuje sběr logů, jejich transformaci a následné ukládání na zvolené místo. Může být použit samostatně, bez ostatních nástrojů Elastic Stacku, nicméně nejčastěji je použit právě v kombinaci s Elasticsearch. Nejčastěji se můžete setkat s označením **ELK** - Elasticsearch, Logstash a Kibana.

Logstash disponuje řadou pluginů, s jejichž pomocí se umí připojit na nejrůznější zdroje dat - soubory, databáze nebo fronty.

Nevýhodou Logstash je jeho hardwarová náročnost - ke svému běhu potřebuje JVM, což je především paměťově náročné. Navíc, pokud by měl sbírat logy z více serverů, je nutné ho na všechny servery nainstalovat, což jeho výslednou paměťovou náročnost zvyšuje.

Návod na instalaci Logstash: <https://www.elastic.co/guide/en/logstash/current/installing-logstash.html>

Logstash po spuštění zpracovává všechny nakonfigurované **pipelines**. Ty mohou běžet neustále, mohou být spouštěny v zadané časy, nebo být spuštěny pouze jednou. Každá pipeline se skládá ze tří částí:



Nastavení pipeline sestává z vytvoření `.conf` souboru s definovanými sekcemi:

```
input {
  file {
    path => ["/tmp/sample.log"]
    start_position => "beginning"
    sincedb_path => "/dev/null"
  }
}

filter {
  csv {
    separator => ","
    columns => ["id", "title", "description"]
  }
}
```

```

    }
}

output {
  elasticsearch {
    hosts => ["http://localhost:9200"]
  }
}

```

V konfiguraci je definovaný jako vstupní soubor `/tmp/sample.log`. Otevřete jej v editoru a přidejte do něj libovolný obsah:

```
nano /tmp/sample.log
```

Logstash můžeme spustit z konzole následujícím příkazem:

```
bin/logstash -f cesta/nazev-konfiguracniho-souboru.conf
```

V tomto případě spustíme Logstash na popředí, přičemž nám zpracovává jedinou pipeline. Pokud jich cheme mít spuštěných více naráz, je nutné nainstalovat a spustit Logstash jako službu. Místo, kam se všechny pipeline ukládají pak záleží na daném operačním systému, například u Linuxu Debian to bude `/etc/logstash/conf.d/`.

Výše uvedená konfigurace odesílá data do lokálního Elasticsearch. Pro napojení na Elastic Cloud využijte [Cloud ID](#):

```

output {
  elasticsearch {
    cloud_id => "My_deployment:abdhbdsbdjhasbdjabdjas==",
    cloud_auth => "username:password"
  }
}

```

Co se týče **vstupních pluginů**, dostupné jsou například:

- Výše uvedený `file` čte data z textového souboru
- `http` získává data pomocí HTTP requestů
- `imap` se dokáže napojovat na e-mailové schránky
- `jdbc` se dokáže připojit do databáze
- `tcp` a `udp` otevře port, na kterém přijímá requesty
- kompletní seznam vstupních pluginů naleznete v [dokumentaci](#)

V případě **výstupních pluginů** jsou nejčastěji používány:

- `elasticsearch`, který data ukládá data do elasticsearch

- `stdout` používá se pro debugging - výpis do terminálu
- a další, viz opět [dokumentace](#); je možné data ukládat obdobným způsobem, jako je přijímat - do databází, souborů, na zadané URL, vysílat eventy

Co se **filtrů** týče, jde převážně o [pluginy](#) pro parsování nejrůznějších vstupních formátů. Dále je lze využít pro modifikaci dat nebo jejich obohacování dohledáváním v jiných databázích nebo v Elasticsearch.

## Beats

Beats jsou oproti Elasticsearch malé, relativně jednoduché "skripty" pro sběr dat a jejich odesílání do Elasticsearch, případně Logstash. Jsou distribuovány jako binární soubory bez dalších závislostí.

Beats umí komunikovat jak s Elasticsearch, tak s Kibanou. Po spuštění si tak dokážou nastavit mapping v Elasticsearch i vytvořit základní dashboardy v Kibaně.

Do skupiny Beats patří řada jednotlivých nástrojů, přičemž každý slouží k sběru jiných dat:

### Filebeat

Čte data z souborů, například aplikační logy. Zjednodušeně řečeno dělá to, co `tail -f` a výstup ukládá do Elasticsearch. Disponuje několika předpřipravenými vstupními formáty souborů, například pro Apache, NGINX, MySQL nebo i samotný Elasticsearch.

Filebeat nainstalujeme (zde na Debian/Ubuntu server) příkazem:

```
apt install filebeat
```

Konfigurace je dostupná otevřením souboru:

```
nano /etc/filebeat/filebeat.yml
```

Pokud bychom chtěli dosáhnout stejného chování jako u logstash, bylo by to možné pomocí následující konfigurace:

```
filebeat.inputs:
- type: log
  paths:
  - /tmp/filebeat-example.log

output.elasticsearch:
  hosts: ["http://localhost:9200"]
```

V případě použití Elastic Cloudu je třeba nastavit výstup odlišně - nastavíte jej na nejvyšší úrovni konfigurace (a přepíše tak automaticky nastavení `output.elasticsearch`):

```
cloud.id: "My_deployment:dsjhfjdfshfsuhfis=="
cloud.auth: "username:password"
```

Filebeat je možné spustit obdobně jako Logstash:

```
systemctl enable filebeat
systemctl start filebeat
```

Výhodou využití beats je, že obsahují předpřipravené konfigurace pro často používané aplikace. A to nejen pro zpracování jejich logů, ale také dashboardy v Kibaně. Například log webového serveru NGINX, který vypadá následovně:

```
93.180.71.3 - - [17/May/2015:08:05:32 +0000] "GET /downloads/product_1 HTTP/1.1" 304 0
 "-" "Debian APT-HTTP/1.3 (0.8.16~exp12ubuntu10.21)"
```

Pro aktivaci zpracování těchto logů je třeba zapnout podporu pro odpovídající modul:

```
filebeat modules enable nginx
```

Alternativně lze daný modul uvést jako argument při spuštění beats, nebo v konfiguračním souboru `filebeat.modules`.

Modul je aktivován s výchozí konfigurací, kterou lze podle potřeby upravit. Například můžeme změnit lokalitu logů, které má Filebeat sledovat, a to editací souboru `modules.d/nginx.yml`:

```
- module: nginx
  access:
    enabled: true
    var.paths: ["/path/to/nginx/access.log*"]
  error:
    enabled: true
    var.paths: ["/path/to/nginx/error.log*"]
```

Po aktivaci požadovaných modulů je třeba provést výchozí nastavení - vytvoření mappingu v Elasticsearch a dashboardů v Kibaně:

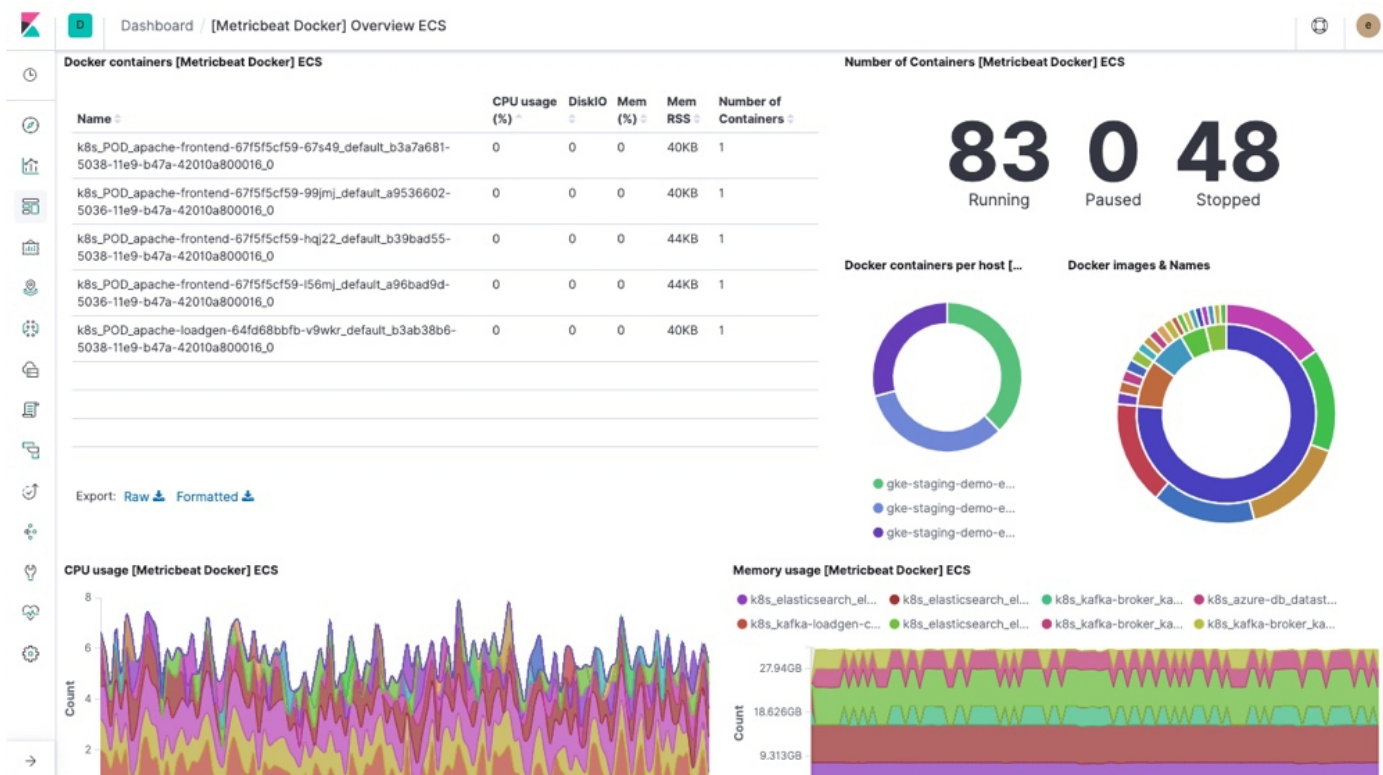
```
/usr/share/filebeat/bin/filebeat setup -e
```

Jakmile začnou do logů přibývat nové záznamy, je možné je sledovat přímo v Kibaně:



## Metricbeat

Metricbeat sbírá metriky o využití systémových zdrojů. S jeho pomocí je pak možné sledovat využití CPU, paměti a další důležité metriky. Kromě samotných serverů jej lze použít i pro monitoring aplikací (např. MongoDB). V neposlední řadě dokáže sledovat linuxové kontejnery, ať už jde o Docker samotný, nebo o orchestraci kontejnerů pomocí Kubernetes:



## Packetbeat

Packetbeat sbírá data o síťovém provozu. S jeho pomocí lze také sledovat provoz databází. Takto například vypadá dashboard pro sledování HTTP provozu:



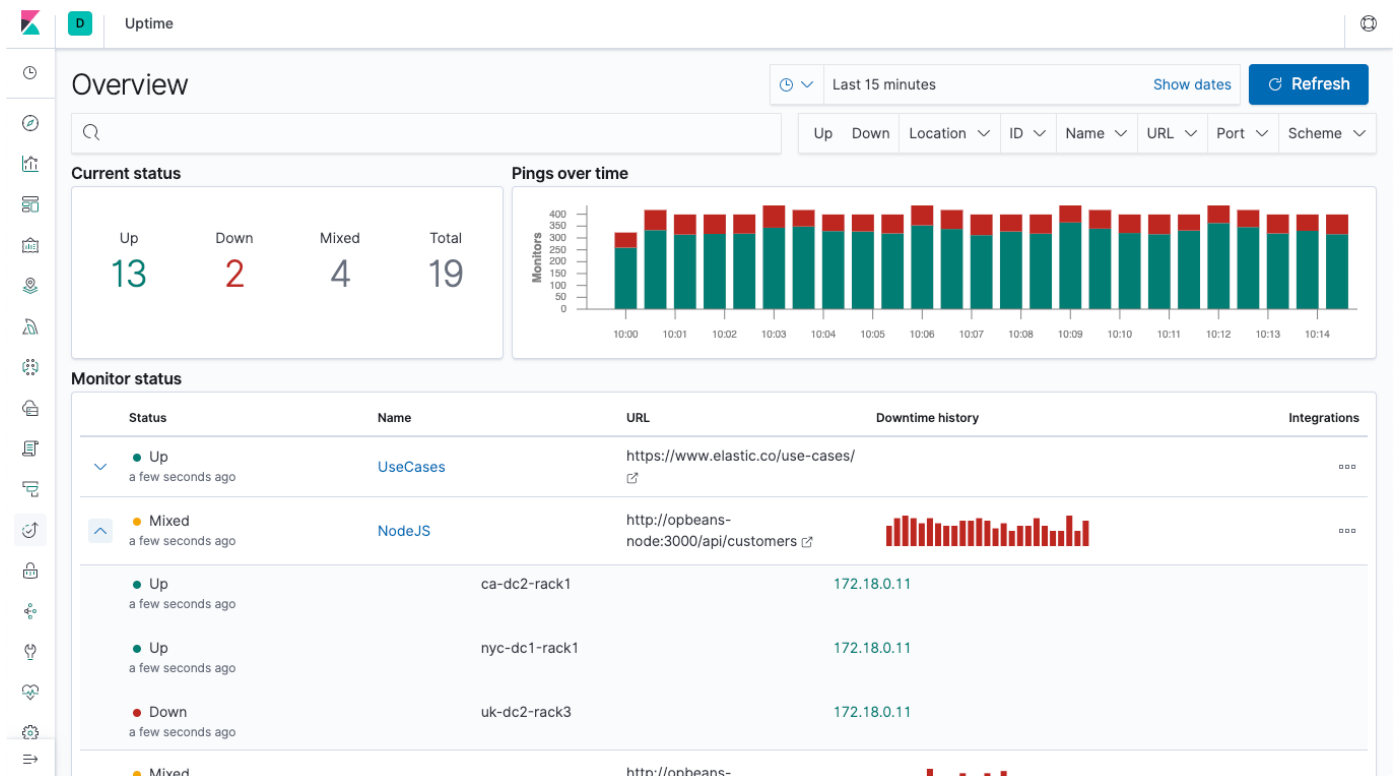
## Winlogbeat

Pokud provozujete na serverech operační systém MS Windows, bude užitečný Winlogbeat, který čte veškeré systémové události v Elasticsearch a přeposílá je do Elasticsearch. Může jít o přihlašování uživatelů, systémové chyby nebo události týkající se HW, například připojování disků.



## Heartbeat

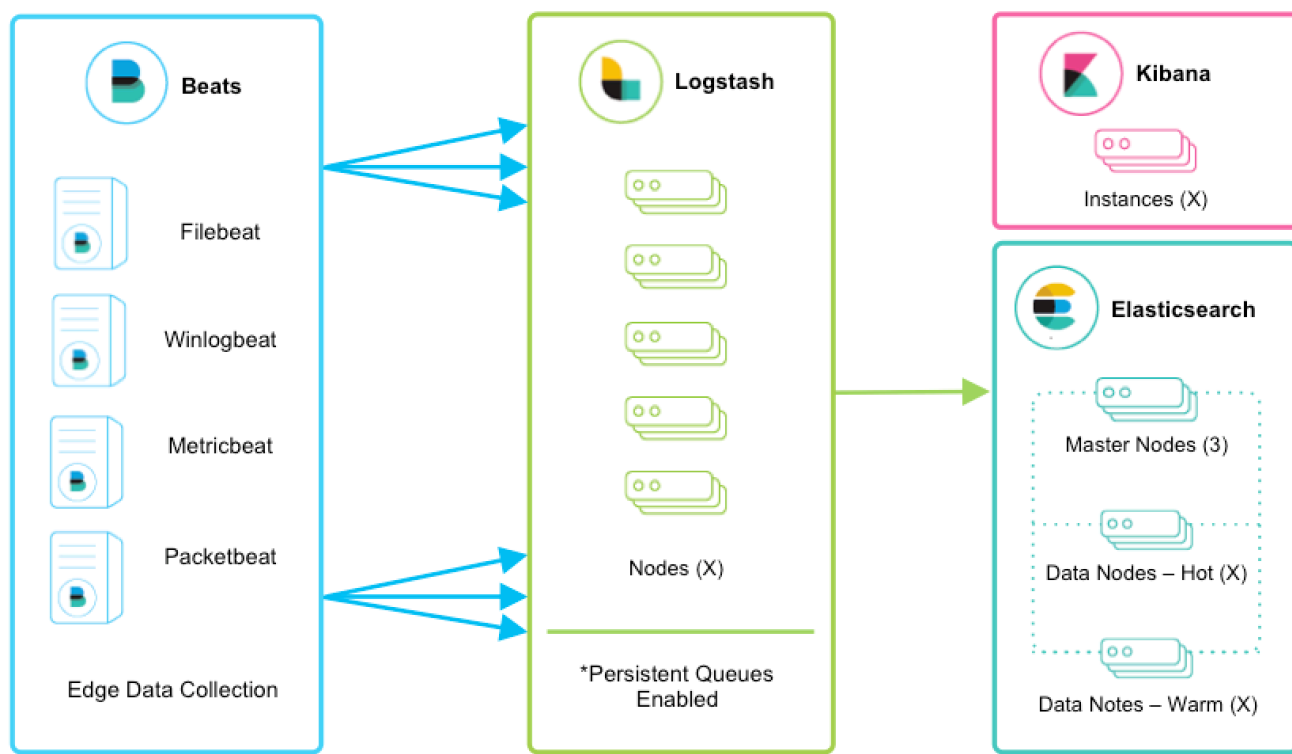
Pokud chcete sledovat dostupnost nejrůznějších systémů, lze k tomu využít Heartbeat. Jde ve své podstatě o náhradu online služeb typu pingdom nebo uptime.





## Logstash a Beats

Často dává smysl používat Beats společně s Logstash. Pokud bychom měli příliš mnoho běžících Beats agentů, kteří se naráz připojují do Elasticsearch, nemusí to být spolehlivé ani efektivní. V takovou chvíli je vhodné vložit Logstash mezi Elasticsearch a Beats:



Pokud Beats posílají výstup do více nodů (ať už Elasticsearch nebo Logstash), je vhodné nastavit [load balancing](#). Díky tomu budou všechny cíloné uzly optimálně vytíženy.

Pro propojení Beats a Logstash je třeba nejprve nastavit výstup v Beats. Například v `filebeat.yml`:

```
output.logstash:
  hosts: ["localhost:5044"]
```

A následně také nastavit vstup Logstash tak, aby rozuměl výše uvedenému výstupu v `.conf` souboru odpovídajícímu dané pipeline:

```
input {
  beats {
    port => "5044"
  }
}
```

Funkce dohledávání (enrichment), které se dějí v sekci filter v Logstash lze nahradit tzv. **Ingest Pipeline** v Elasticsearch.



# Kibana

Kibana neslouží jen ke spouštění dotazů a managementu clusteru. Její hlavní úlohou je vizualizace dat v Elasticsearch a jejich prohledávání pomocí grafického rozhraní.

