

NOV 07, 2019

Migrating from Nagios to Prometheus

Runtastic Infrastructure

Base

Linux (Ubuntu)

SDN (Cisco)

Chef

Terraform



Virtualization

Linux KVM

OpenNebula

3600 CPU Cores

20 TB Memory

100 TB Storage



Core DBs

Physical

Hybrid

Big



Technologies

Really a lot
open source



Our Monitoring back in 2017...

- Nagios
 - Many Checks for all Servers
 - Checks for NewRelic
- Pingdom
 - External HTTP Checks
 - Specific Nagios Alerts
 - Alerting via SMS
- NewRelic
 - Error Rate
 - Response Time

Configuration hell....

```
"Mongodb Shard 01 Repl 01 Replication Lag": {
  "command": "check_mongodb!prd-core-mongodb-s01r01.runtastic.com!replication_lag!27018!3!5",
  "service_template": "generic-service"
},
"Mongodb Shard 01 Repl 01 Replication State": {
  "command": "check_mongodb!prd-core-mongodb-s01r01.runtastic.com!replset_state!27018!0!0",
  "service_template": "generic-service"
},
"Mongodb Shard 01 Repl 01 command per Sec": {
  "command": "check_mongodb_query!prd-core-mongodb-s01r01.runtastic.com!queries_per_second!27018!900!1200!command",
  "service_template": "generic-service"
},
"Mongodb Shard 01 Repl 01 getmore per Sec": {
  "command": "check_mongodb_query!prd-core-mongodb-s01r01.runtastic.com!queries_per_second!27018!750!950!getmore",
  "service_template": "generic-service"
},
"Mongodb Shard 01 Repl 01 insert per Sec": {
  "command": "check_mongodb_query!prd-core-mongodb-s01r01.runtastic.com!queries_per_second!27018!300!400!insert",
  "service_template": "generic-service"
},
"Mongodb Shard 01 Repl 01 query per Sec": {
  "command": "check_mongodb_query!prd-core-mongodb-s01r01.runtastic.com!queries_per_second!27018!750!950!query",
  "service_template": "generic-service"
},
"Mongodb Shard 01 Repl 01 update per Sec": {
  "command": "check_mongodb_query!prd-core-mongodb-s01r01.runtastic.com!queries_per_second!27018!300!400!update",
  "service_template": "generic-service"
},
"Mongodb Shard 01 Repl 02 Index Miss Ratio": {
  "command": "check_mongodb!prd-core-mongodb-s01r02.runtastic.com!index_miss_ratio!27018!.005!.01",
  "service_template": "generic-service"
},
```

Alert overflow...

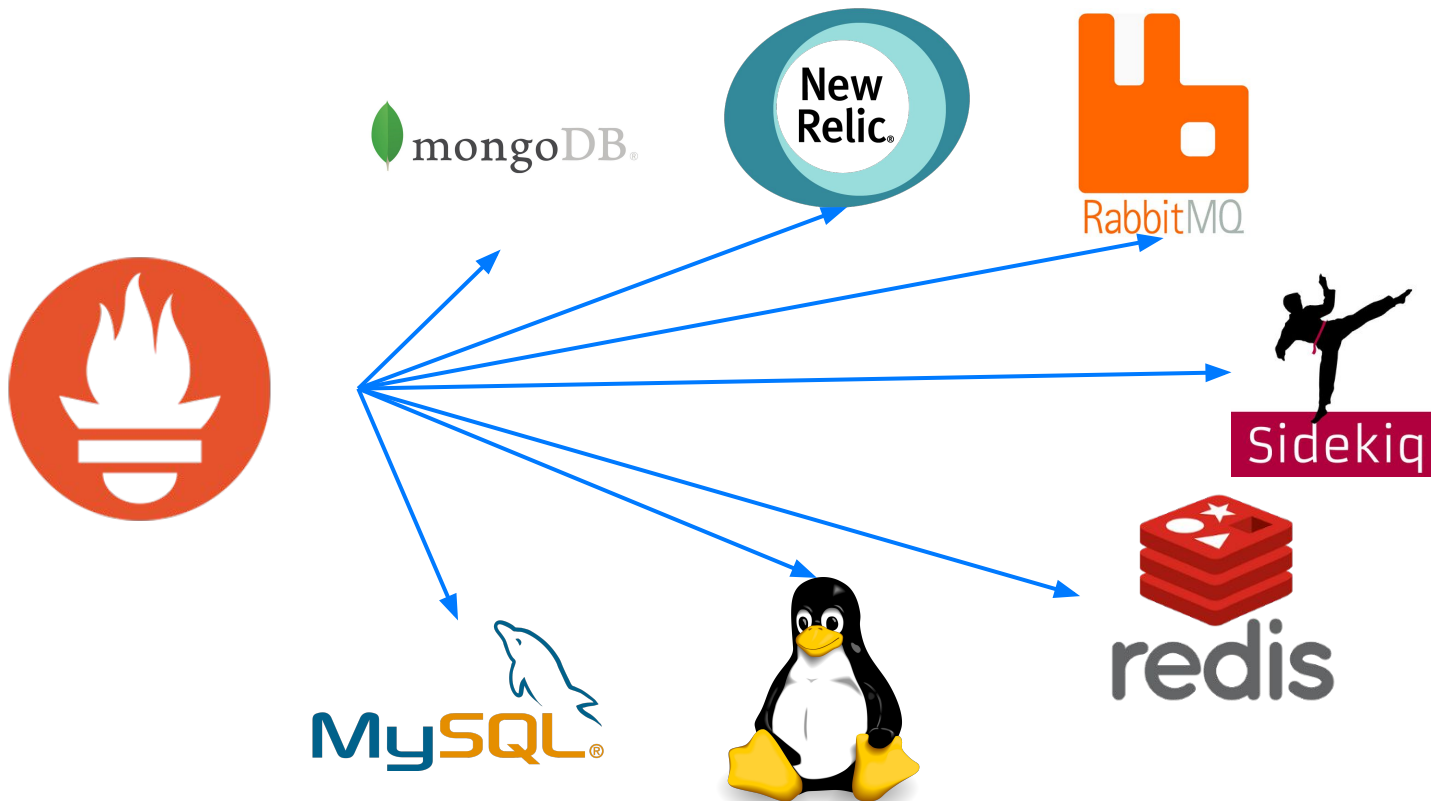
nagstamon 3.0.2							Go to monitor...	Filters	Recheck all	Refresh	Settings		
prd-photos-worker-466.runtastic.com	OOM killer	CRITICAL	2019-03-24 10:24:00	5d 19h 10m 10s	4/4	OOM Killer found!							
prd-polarexternals-haproxy-002.runtastic.com	HAProxy	CRITICAL	2019-03-24 10:30:36	31d 23h 43m 8s	4/4	Active 'tf-prd-polarexternals-server-002' is DOWN on 'tf-polarexternals-trinid							
prd-samples-server-001.runtastic.com	Current Load	CRITICAL	2019-03-24 10:46:17	0d 0h 55m 11s	4/4	CRITICAL - load average: 11.81, 11.84, 11.10							
prd-samples-server-002.runtastic.com	Current Load	CRITICAL	2019-03-24 10:45:23	0d 0h 56m 3s	3/4	CRITICAL - load average: 10.86, 10.72, 10.27							
prd-samples-server-003.runtastic.com	Current Load	CRITICAL	2019-03-24 10:43:35	0d 0h 31m 46s	3/4	CRITICAL - load average: 12.05, 11.29, 10.24							
prd-samples-server-004.runtastic.com	Current Load	CRITICAL	2019-03-24 10:31:23	0d 0h 43m 25s	3/4	CRITICAL - load average: 9.44, 9.29, 8.77							
prd-samples-server-005.runtastic.com	Current Load	CRITICAL	2019-03-24 10:48:06	0d 0h 53m 30s	4/4	CRITICAL - load average: 11.08, 10.84, 10.12							
prd-samples-server-006.runtastic.com	Current Load	CRITICAL	2019-03-24 10:43:12	0d 0h 58m 2s	3/4	CRITICAL - load average: 14.47, 12.01, 10.89							
prd-samples-server-007.runtastic.com	Current Load	CRITICAL	2019-03-24 10:32:33	0d 0h 16m 11s	3/4	CRITICAL - load average: 8.41, 8.56, 8.46							
prd-samples-server-008.runtastic.com	Current Load	CRITICAL	2019-03-24 10:28:05	0d 0h 46m 43s	3/4	CRITICAL - load average: 9.91, 9.30, 8.62							
prd-samples-server-009.runtastic.com	Current Load	CRITICAL	2019-03-24 10:45:02	0d 0h 56m 22s	3/4	CRITICAL - load average: 10.21, 10.78, 10.07							
prd-samples-server-010.runtastic.com	Current Load	CRITICAL	2019-03-24 10:39:42	0d 1h 1m 31s	3/4	CRITICAL - load average: 10.21, 10.03, 9.63							
prd-samples-server-011.runtastic.com	Current Load	CRITICAL	2019-03-24 10:20:24	0d 1h 19m 51s	4/4	CRITICAL - load average: 14.47, 13.05, 12.20							
prd-samples-server-012.runtastic.com	Current Load	CRITICAL	2019-03-24 10:22:52	0d 0h 25m 52s	3/4	CRITICAL - load average: 7.95, 8.55, 8.16							
prd-samples-server-013.runtastic.com	Current Load	CRITICAL	2019-03-24 10:45:25	0d 0h 56m 2s	3/4	CRITICAL - load average: 12.98, 10.97, 9.87							
prd-samples-server-014.runtastic.com	Current Load	CRITICAL	2019-03-24 10:37:53	0d 1h 3m 24s	3/4	CRITICAL - load average: 10.28, 9.91, 9.60							
prd-samples-server-015.runtastic.com	Current Load	CRITICAL	2019-03-24 10:32:07	0d 0h 42m 43s	2/4	CRITICAL - load average: 8.91, 9.71, 9.30							
prd-samples-server-016.runtastic.com	Current Load	CRITICAL	2019-03-24 10:33:00	0d 0h 41m 52s	3/4	CRITICAL - load average: 10.58, 10.17, 9.65							
prd-samples-server-017.runtastic.com	Current Load	CRITICAL	2019-03-24 10:43:13	0d 0h 32m 7s	3/4	CRITICAL - load average: 13.07, 11.05, 10.10							
prd-samples-server-018.runtastic.com	Current Load	CRITICAL	2019-03-24 10:33:59	0d 0h 40m 56s	3/4	CRITICAL - load average: 9.49, 9.80, 9.36							
prd-samples-server-019.runtastic.com	Current Load	CRITICAL	2019-03-24 10:23:25	0d 0h 25m 19s	2/4	CRITICAL - load average: 10.59, 10.41, 9.25							
prd-samples-server-020.runtastic.com	Current Load	CRITICAL	2019-03-24 10:43:51	0d 0h 57m 26s	3/4	CRITICAL - load average: 13.77, 11.29, 10.42							
prd-samples-server-021.runtastic.com	Current Load	CRITICAL	2019-03-24 10:39:42	0d 0h 35m 33s	3/4	CRITICAL - load average: 10.65, 10.75, 10.40							
prd-samples-server-022.runtastic.com	Current Load	CRITICAL	2019-03-24 10:24:32	0d 1h 15m 51s	3/4	CRITICAL - load average: 12.87, 12.63, 11.35							
prd-samples-server-023.runtastic.com	Current Load	CRITICAL	2019-03-24 10:48:07	0d 0h 53m 29s	4/4	CRITICAL - load average: 13.95, 13.81, 12.13							
prd-samples-server-024.runtastic.com	Current Load	CRITICAL	2019-03-24 10:36:01	0d 0h 39m 5s	3/4	CRITICAL - load average: 10.76, 10.94, 10.26							
prd-samples-worker-005.runtastic.com	OOM killer	CRITICAL	2019-03-24 10:18:46	40d 22h 45m 43s	4/4	OOM Killer found!							
prd-samples-worker-006.runtastic.com	Current Load	CRITICAL	2019-03-24 10:18:41	0d 1h 47m 2s	4/4	CRITICAL - load average: 4.55, 4.62, 4.67							
prd-ssl-appws-001.runtastic.com	HTTPS connections	CRITICAL	2019-03-24 10:45:58	0d 2h 23m 29s	4/4	80950 connections on port 443!							
prd-ssl-appws-001.runtastic.com	Current Load	CRITICAL	2019-03-24 10:48:07	0d 0h 57m 29s	4/4	CRITICAL - load average: 4.51, 4.38, 4.58							
prd-ssl-hubs-002.runtastic.com	HTTPS connections	CRITICAL	2019-03-24 10:31:43	0d 1h 42m 24s	4/4	59969 connections on port 443!							
prd-ssl-hubs-002.runtastic.com	Current Load	CRITICAL	2019-03-24 10:29:42	0d 0h 45m 4s	3/4	CRITICAL - load average: 5.06, 4.75, 4.64							
prd-ssl-services-001.runtastic.com	Bandwidth eth0	CRITICAL	2019-03-24 10:45:29	0d 0h 3m 15s	1/4	eth0 total: 37850 KB/s tx: 18857 KB/s rx: 18993 KB/s							
prd-test-rabbitmq-001.runtastic.com	Rabbitmq Watermark	CRITICAL	2019-03-24 10:20:53	18d 17h 57m 4s	4/4	RABBITMQ_WATERMARK CRITICAL - Received 500 Can't connect to prd-test-							

Goals for our new Monitoring system

- Make On Call as comfortable as possible
- Automate as much as possible
- Make use of graphs
- Rework our alerting
- Make it scaleable!

Starting with Prometheus...

Prometheus



Our Prometheus Setup



- 2x Bare Metal
- 8 Core CPU
- Ubuntu Linux
- **7.5 TB** of Storage
- **7 month** of Retention time
- Internal TSDB

Automation

Our Goals for Automation

- Roll out Exporters on new servers automatically
 - using Chef
- Use Service Discovery in Prometheus
 - using Consul
- Add HTTP Healthcheck for a new Microservice
 - using Terraform
- Add Silences with 30d duration
 - using Terraform

Consul

- Consul for our Terraform State
- Agent Rollout via Chef
- One Service definition per Exporter on each Server

Consul

prd-sharing-server-001

 10.210.100.11

Health Checks **Services** Round Trip Time Lock Sessions Meta Data

Search by name/port



Service	Port	Tags
prd-sharing-server-001-mongodbboxporter	9216	prometheus role:trinidad service:sharing exporter:mongodb
prd-sharing-server-001-nodebboxporter	9100	prometheus role:trinidad service:sharing exporter:node
prd-sharing-server-001-sidekiqbboxporter	9998	prometheus role:trinidad service:sharing exporter:sidekiq

What Labels do we need?

- What's the Load of all workers of our Newsfeed service?
 - `node_load1{service="newsfeed", role="workers"}`
- What's the Load of a specific Leaderboard server?
 - `node_load1{hostname="prd-leaderboard-server-001"}`

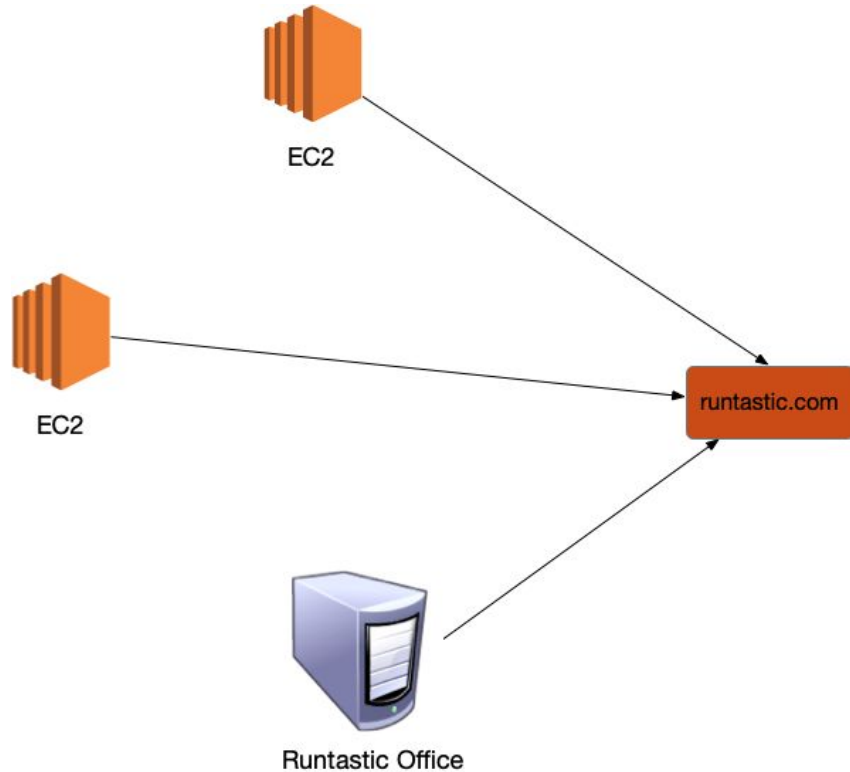
...and how we implemented them in Consul

```
{  
  "service": {  
    "name": "prd-sharing-server-001-mongodbexporter",  
    "tags": [  
      "prometheus",  
      "role:trinidad",  
      "service:sharing",  
      "exporter:mongodb"  
    ],  
    "port": 9216  
  }  
}
```

Scrape Configuration

```
- job_name: prd
  consul_sd_configs:
    - server: 'prd-consul:8500'
      token: 'ourconsultoken'
      datacenter: 'lnz'
  relabel_configs:
    - source_labels: [__meta_consul_tags]
      regex: .*,prometheus,.*
      action: keep
    - source_labels: [__meta_consul_node]
      target_label: hostname
    - source_labels: [__meta_consul_tags]
      regex: .*,service:([^\,]+),.*
      replacement: '${1}'
      target_label: service
```


External Health Checks



- 3x Blackbox Exporters
- Accessing SSL Endpoints
- Checks for
 - HTTP Response Code
 - SSL Certificate
 - Duration

Add Healthcheck via Terraform

```
resource "consul_service" "health_check" {  
  name    = "${var.srv_name}-healthcheck"  
  node    = "blackbox_aws"  
  
  tags = [  
    "healthcheck",  
    "url:https://status.runtastic.com/${var.srv_name}",  
    "service:${var.srv_name}",  
  ]  
}
```

Job Config for Blackbox Exporters

```
- job_name: blackbox_aws
  metrics_path: /probe
  params:
    module: [http_health_monitor]
  consul_sd_configs:
  - server: 'prd-consul:8500'
    token: 'ourconsultoken'
    datacenter: 'lnz'
  relabel_configs:
    - source_labels: [__meta_consul_tags]
      regex: .*,healthcheck,.*
      action: keep
    - source_labels: [__meta_consul_tags]
      regex: .*,url:([^,]+),.*
      replacement: '${1}'
      target_label: __param_target
```

Add Silence via Terraform

```
resource "null_resource" "prometheus_silence" {

  provisioner "local-exec" {
    command = <<EOF
      ${var.amtool_path} silence add 'service=~SERVICENAME'
\
      --duration='30d' \
      --comment='Silence for the newly deployed service' \
      --alertmanager.url='http://prd-alertmanager:9093'
    EOF
  }
}
```


OpsGenie

Our Initial Alerting Plan

- Alerts with Low Priority
 - Slack Integration
- Alerts with High Priority (OnCall)
 - Slack Integration
 - OpsGenie

...why not forward all Alerts to OpsGenie?

Define OpsGenie Alert Routing

- **Prometheus OnCall** Integration
 - High Priority Alerts (e.g. Service DOWN)
 - Call the poor On Call Person
 - Post Alerts to Slack #topic-alerts
- **Prometheus Ops** Integration
 - Low Priority Alerts (e.g. Chef-Client failed runs)
 - Disable Notifications
 - Post Alerts to Slack #prometheus-alerts

Setup Alertmanager Config

- receiver: 'opsgenie_oncall'
 group_wait: 10s
 group_by: ['...']
 match:
 oncall: 'true'
- receiver: 'opsgenie'
 group_by: ['...']
 group_wait: 10s

...and its receivers

```
- name: "opsgenie_oncall"
  opsgenie_configs:
    - api_url: "https://api.eu.opsgenie.com/"
      api_key: "ourapitoken"
      priority: "{{ range .Alerts }}{{ .Labels.priority }}{{ end
    }}"
      message: "{{ range .Alerts }}{{ .Annotations.title }}{{ end
    }}"
      description: "{{ range .Alerts }}\n{{ .Annotations.summary
    }}\n\n{{ if ne .Annotations.dashboard \"\" -}}\nDashboard:\n{{
    .Annotations.dashboard }}\n{{- end }}{{ end }}"
      tags: "{{ range .Alerts }}{{ .Annotations.instance }}{{ end
    }}"
```

Why we use `group_by['...']`

- Alert Deduplication from OpsGenie
- Alerts are being grouped
- Overlook Alerts

Example Alerting Rule for On Call

```
- alert: HTTPProbeFailedMajor
  expr: max by(instance,service) (probe_success) < 1
  for: 1m
  labels:
    oncall: "true"
    priority: "P1"
  annotations:
    title: "{{ $labels.service }}" DOWN"
    summary: "HTTP Probe for {{ $labels.service }}"
    FAILED.\nHealth Check URL: {{ $labels.instance }}"
```

Example Alerting Rule with Low Priority

```
- alert: MongoDB-ScannedObjects
  expr: max by(hostname,
service) (rate(mongodb_mongod_metrics_query_executor_total[30m])) >
500000
  for: 1m
  labels:
    priority: "P3"
  annotations:
    title: "MongoDB - Scanned Objects detected on {{
$labels.service }}"
    summary: "High value of scanned objects on {{
$labels.hostname }} for service {{ $labels.service }}"
    dashboard:
"https://prd-prometheus.runtastic.com/d/oCziI1Wmk/mongodb"
```

Alert Management via Slack



Opsgenie APP 12:09

#2495: [Prometheus]: ACI - Node Health Score LOW

Health Score of Node inf-net-leaf-102 is at 90.

Unacknowledge

Close

Other actions...



OpsGenie EU APP 08:30

Niko Dominkowitsch acknowledged alert #2495 "[Prometheus]: ACI - Node Health Score LOW"



OpsGenie EU APP 09:55

Niko Dominkowitsch added note "Ticket opened @ Kapsch" to alert #2495 "[Prometheus]: ACI - Node Health Score LOW"

Setting up the Heartbeat

```
groups:
- name: opsgenie.rules
  rules:
- alert: OpsGenieHeartBeat
  expr: vector(1)
  for: 5m
  labels:
    heartbeat: "true"
  annotations:
    summary: "Heartbeat for OpsGenie"
```

...and its Alertmanager Configuration

```
- receiver: 'opsgenie_heartbeat'
  repeat_interval: 5m
  group_wait: 10s
  match:
    heartbeat: 'true'

- name: "opsgenie_heartbeat"
  webhook_configs:
    - url:
        'https://api.eu.opsgenie.com/v2/heartbeats/prd_prometheus/ping'
      send_resolved: false
      http_config:
        basic_auth:
          password: "opsgenieAPIkey"
```


CI/CD Pipeline

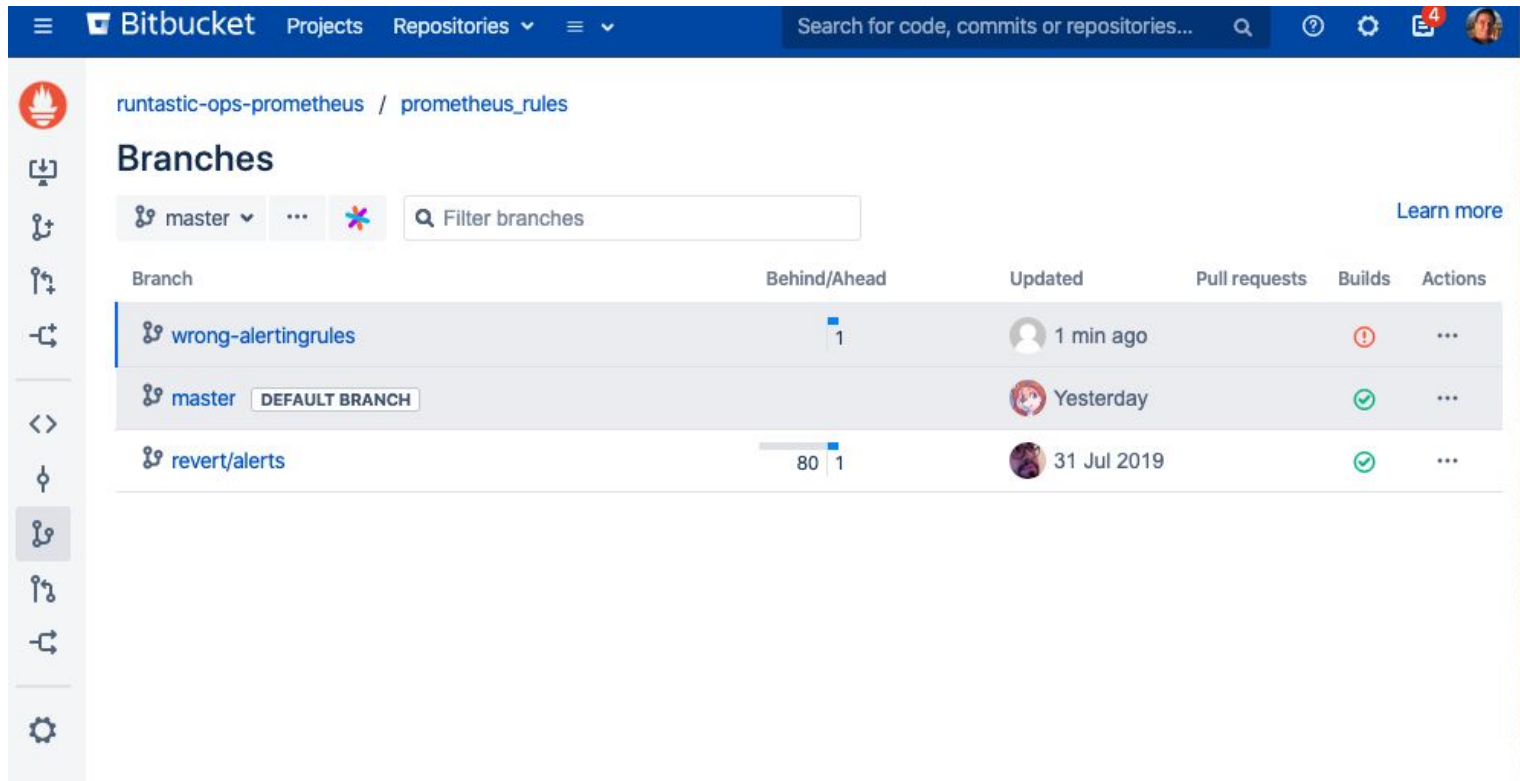
Goals for our Pipeline

- Put all Alerting and Recording Rules into a Git Repository
- Automatically test for syntax errors
- Deploy **master** branch on all Prometheus servers
- Merge to master —> Deploy on Prometheus

How it works

- Jenkins
 - running promtool against each .yaml file
- Bitbucket sending HTTP calls when master branch changes
- Ruby based HTTP Handler on Prometheus Servers
 - Accepting HTTP calls from Bitbucket
 - Git pull
 - Prometheus reload

Verify Builds for each Branch



The screenshot shows the Bitbucket web interface for a repository named 'runtastic-ops-prometheus' with the path 'prometheus_rules'. The 'Branches' section is active, displaying a table of branches. The 'master' branch is the default. The 'wrong-alertingrules' branch is 1 commit behind 'master' and has a failed build (indicated by a red exclamation mark). The 'revert/alerts' branch is 80 commits behind 'master' and has a successful build (indicated by a green checkmark).

Bitbucket Projects Repositories Search for code, commits or repositories...

runtastic-ops-prometheus / prometheus_rules

Branches

master Filter branches Learn more

Branch	Behind/Ahead	Updated	Pull requests	Builds	Actions
wrong-alertingrules	1	1 min ago		❌	...
master DEFAULT BRANCH		Yesterday		✅	...
revert/alerts	80 1	31 Jul 2019		✅	...



THANK YOU



runtastic.com



Niko Dominkowitsch
Infrastructure Engineer

niko.dominkowitsch@runtastic.com