



SPANISH

**ZABBIX** 

MEETUP ONLINE '20

July 30, 2020



- todos los micrófonos están silenciados para evitar sonidos e interrupciones innecesarias
- por favor hacé tus preguntas en la sección Q&A, no en el chat
- usá el chat para discusiones, networking o aplausos
- usa el hashtag **#ZabbixMeetupOnline** si posteas sobre el evento en tus redes sociales



**ZABBIX** 5.0

ZABBIX AGENT2  
LA EVOLUCIÓN DEL AGENTE ZABBIX

**CUSTOS**  
MONITORING



# ZABBIX AGENT2

## LA NUEVA GENERACIÓN

- ☑ Por qué un nuevo agente ?
  - Dificultades con el agente tradicional
    - Conexiones persistentes
    - Mantener el estado entre chequeos
    - Recibir traps
    - El lenguaje de programación
    - Chequeos que necesitan tiempo

# ZABBIX AGENT2

## LA NUEVA GENERACIÓN

- ⊗ Escrito en Go Lang ( versión 1.13 o posterior )
- ⊗ Infraestructura para plugins
- ⊗ Desde la versión 4.2 esta en pruebas, ahora en producción
- ⊗ Todas las “viejas” key soportadas
  - Nuevas key exclusivas

# ZABBIX AGENT2 - LA NUEVA GENERACIÓN

- ☑ Timeouts implementados a nivel de los plugins
  - Ahora podemos esperar sin entorpecer
- ☑ Active checks
  - Corren en paralelo ( ej: lectura de logs )
  - Intervalos flexibles ( Scheduled/Flexible )
- ☑ Menos conexiones TCP
- ☑ Acepta traps y eventos (MQTT subscribe, escucha en puertos TCP/UDP, etc)
- ☑ Soporta (casi todas) las configuraciones del zabbix agent



# ZABBIX AGENT2 - LA NUEVA GENERACIÓN

## ☑ Monitoreos nativos

- Systemd
- Docker
- Memcached
- MySQL
- PostgreSQL
- Redis





# ZABBIX AGENT2 - COMPONENTES

- ✓ Connector
- ✓ Listener
- ✓ Scheduler

# COMPONENTES

- ☑ **Connector:**
  - Maneja la conexión con el servidor
  - Uno por servidor activo
  - Configuración de los Ítems
  - Buffer de medidas (history cache)



# COMPONENTS

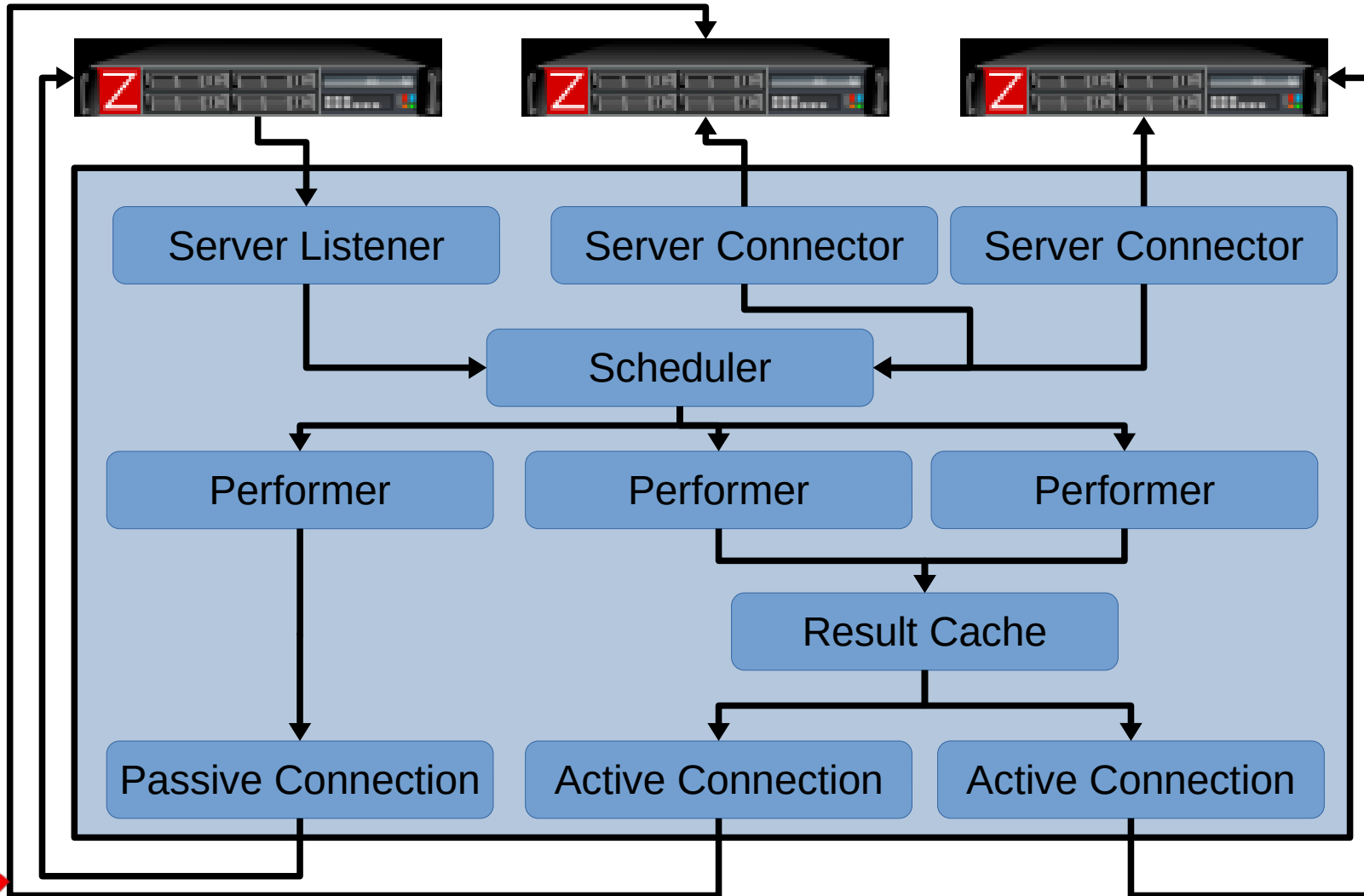
- ☑ Listener
  - Recibe los pedidos de los chequeos pasivos y los pone “en cola” en el scheduler.

# COMPONENTS

- ☑ Scheduler
  - Maneja la cola de pedidos y la concurrencia
  - Uno por agente
- ☑ Dos niveles de cola
  - Cada plugin tiene su propia cola de pedidos
  - El scheduler tiene la cola general de plugins activos



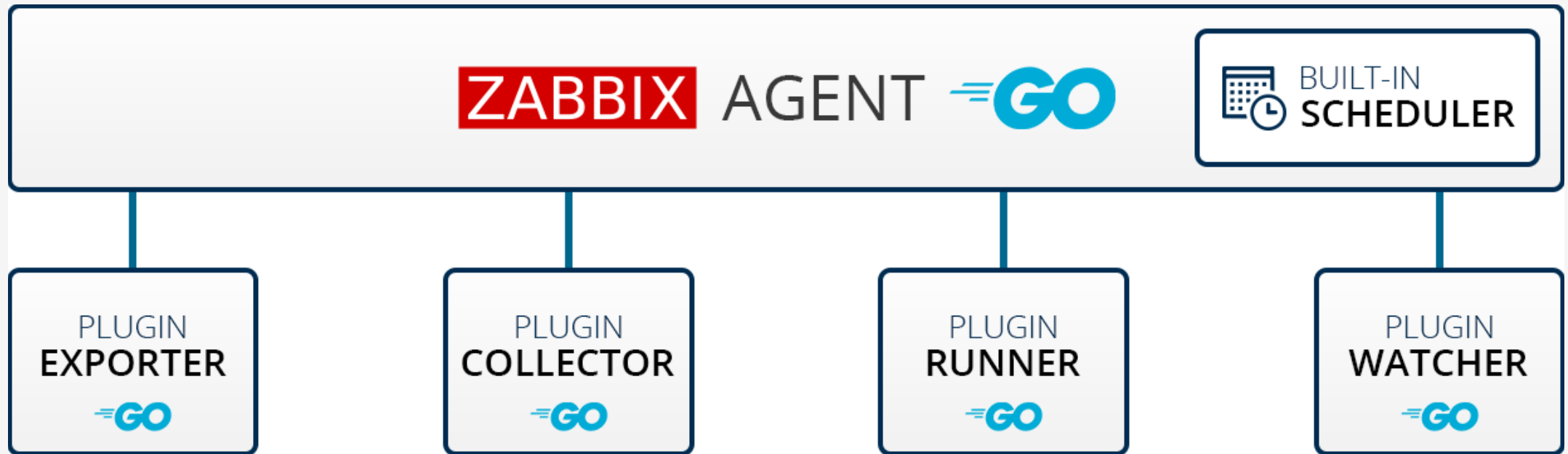
# COMPONENTS





# FRAMEWORK – ESCRIBIENDO UN PLUGIN

- ✓ Fácil como nunca antes
- ✓ Múltiples plugins para distintas tareas



# ZABBIX AGENT 2 - INTERFACES

- ✓ **Exporter**
  - Cosecha valores y retorna nada, un error, uno o múltiples valores
  - Soporta la concurrencia
  
- ✓ **Collector**
  - Cuando se cosecha periódicamente a intervalos regulares, necesita del Exporter
  
- ✓ **Runner**
  - Para inicializar o “limpiar” al inicio y finalización.
  
- ✓ **Watcher**
  - Para tener un método de cosecha “propio”, saltándose el scheduler
  - Escucha en un puerto específico.
  - Para usarse en cosechas de tipo trap
  
- ✓ **Configurator**
  - Configura el plugin con los datos del archivo de configuración



# ZABBIX AGENT2 – MANOS EN LA MASA

- ⊙ **Necesitamos importar el paquete plugin**
  - `import "zabbix.com/pkg/plugin"`
- ⊙ **Definir una estructura y embeber la estructura `plugin.Base`**
  - ```
type Plugin struct {
    plugin.Base
}
var impl Plugin
```
- ⊙ **Implementar una o varias interfaces del plugin**
  - ```
func (p *Plugin) Export(key string, params []string)
(result interface{}, err error) {
    if len(params) > 0 {
        p.Debugf("recibimos %d parametros", len(params))
        return nil, errors.New("Demaciados parametros")
    }
    return time.Now().Format(time.RFC3339)
}
```





# MANOS EN LA MASA

plugin.Exporter

```
type Exporter interface {
    Export(key string, params []string, context
    ContextProvider) (result interface{}, err
    error)
}
```

plugin.Watcher

```
type Watcher interface {
    Watch(requests []*Request, context
    ContextProvider)
}
```

plugin.Collector

```
type Collector interface {
    Collect() error
    Period() int
}
```



# MANOS EN LA MASA

plugin.Runner

```
type Runner interface {  
    Start()  
    Stop()  
}
```

plugin.Configurator

```
type Configurator interface {  
    Configure(globalOptions *GlobalOptions, privateOptions interface{})  
    Validate(privateOptions interface{}) error  
}
```

Registrarse con sus métricas

```
func init() {  
    plugin.RegisterMetrics(&impl, "Time",  
        "system.time", "Returns time string in RFC 3999 format.")  
}
```

Vamos a tener que re compilar el agente para que nuestro plugin funcione



# ZABBIX AGENT2 – EJEMPLO

```
package muySimple
```

```
import "zabbix.com/pkg/plugin"
```

```
type Plugin struct {  
    plugin.Base  
}
```

```
var impl Plugin
```

```
func (p *Plugin) Export(key string, params []string, ctx plugin.ContextProvider)  
    (res interface{}, err error) {  
    // Control de parámetros etc  
    // Conseguimos los datos  
    return  
}
```

```
func init() {  
    plugin.RegisterMetrics(&impl, "muySimple", "key", "Descripción.")  
}
```





# ZABBIX AGENT2 – SIGUIENTES PASOS

- ⊙ Hacer de los plugins unas “librerías dinámicas”
  - Para evitar la compilación del agente
- ⊙ Actualizar las configuraciones en tiempo de ejecución
  - Para evitar para y arrancar el agente
- ⊙ Más plugins
  - Zabbix va a desarrollar nuevos plugins, pero que eso no nos detenga !





Muchas  
gracias !





SPANISH

**ZABBIX** 

MEETUP ONLINE '20

July 30, 2020



- todos los micrófonos están silenciados para evitar sonidos e interrupciones innecesarias
- por favor hacé tus preguntas en la sección Q&A, no en el chat
- usá el chat para discusiones, networking o aplausos
- usa el hashtag **#ZabbixMeetupOnline** si posteas sobre el evento en tus redes sociales