

# Azure App Configuration e Azure KeyVault

## Two store: One configuration

Stefano Mostarda

stefano@aspitalia.com | @sm15455



# Agenda

- Introduzione
- Pattern di utilizzo
- Organizzazione chiavi
- Refresh configurazione
- App Configuration + Key Vault
- Pricing

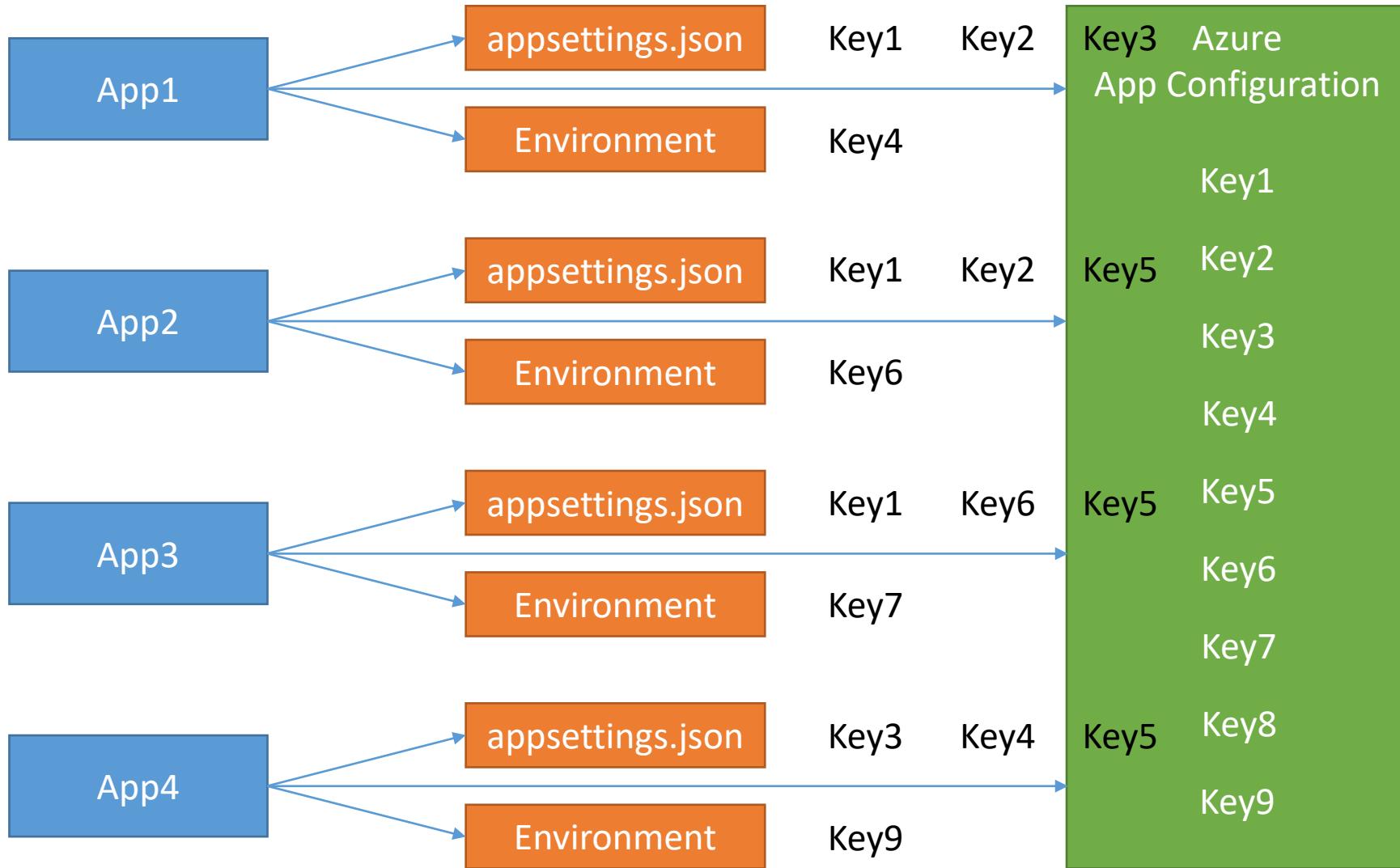


# Azure App Configuration

- Servizio per salvataggio e lettura parametri di configurazione
- Store chiave-valore:
  - Valore primitivo
  - Oggetto JSON
  - Riferimento a secret in Key Vault
- API
- Security:
  - Access key
  - Managed Identity + RBAC
  - Encryption
- Feature flag manager
- In General Availability da dicembre 2020



# Perché Azure App Configuration?



# Perché Azure App Configuration?

- Single source of truth
  - Nessuna duplicazione configurazione
  - Manutenzione più semplice
- Refresh configurazione basato su tempo/attività
- Refresh configurazione on demand
  - Integrazione con EventGrid
- Versioning
- Geo-redundant
  - Integrazione con EventGrid



# App configuration in console

- Package:

```
Microsoft.Extensions.Configuration.AzureAppConfiguration
```

- Startup.cs

```
static void Main(string[] args)
{
    var builder = new ConfigurationBuilder();
    builder.ConfigurationBuilder.AddAzureAppConfiguration(o =>
    {
        o.Connect(new Uri(url), new DefaultAzureCredential());
    });
    var config = builder.Build();
}
```



# App configuration in ASP.NET

- Package:

```
Microsoft.Azure.AppConfiguration.AspNetCore
```

- Program.cs

```
public static IHostBuilder CreateHostBuilder(string[] args) =>
    Host.CreateDefaultBuilder(args)
        .ConfigureWebHostDefaults(webBuilder =>
    {
        webBuilder
            .ConfigureAppConfiguration((hostingContext, config) =>
        {
            var settings = config.Build();
            var credentials = new DefaultAzureCredential();
            config.AddAzureAppConfiguration(o =>
                o.Connect(new Uri(url),
                    new DefaultAzureCredential()));
        })
        .UseStartup<Startup>();
    });
}
```



# App configuration in ASP.NET

- Package:

```
Microsoft.Azure.AppConfiguration.AspNetCore
```

- Program.cs

```
public static IHostBuilder CreateHostBuilder(string[] args) =>
    Host.CreateDefaultBuilder(args)
        .ConfigureWebHostDefaults(webBuilder =>
    {
        webBuilder
            .ConfigureAppConfiguration((hostingContext, config) =>
        {
            var settings = config.Build();
            var credentials = new DefaultAzureCredential();
            config.AddAzureAppConfiguration(o =>
                o.Connect(new Uri(url),
                    new DefaultAzureCredential()));
        })
        .UseStartup<Startup>();
    });
}
```



# Demo

# Key organization

- Chiavi semplici
  - Background
  - BusQueue
  - BusConnectionString
- Chiavi che rappresentano proprietà di un oggetto
  - SMTP:Server
  - SMTP:Port
  - SMTP:Username
  - SMTP:Password
- Chiavi che rappresentano un oggetto tramite JSON
  - SMTPJson

```
{"Server": "smtp.server.com",
  "Port": 20,
  "Username": "user",
  "Password": "pwd"}
```
  - Set content/type to application/json
- Chiavi che referenziano un secret in Key Vault
  - Nomenclatura come chiavi precedenti
  - Valore è url del secret



# Key organization

- Valori multipli per chiave: Label
  - Differentiate values between environments

Key	Label	Value
Background	Dev	Green
Background	Stg	Red
Background	Prod	Blue

- Override configurazione per un'applicazione

Key	Label	Value
Background		Green
Background	ConsoleApp	LightGreen
Background	WebApp	DarkGreen



# Key organization

- SDK consente:
  - Recupero chiavi per prefisso
  - Eliminare prefisso da chiavi
  - Overriding del valore

Key	Label	Value
Background		Green
Background	ConsoleApp	LightGreen
Background	WebApp	DarkGreen



# Key organization

- Instanze per environment
  - Più sicure rispetto utilizzo di label
  - Supporto per RBAC
- Pros:
  - Environment Isolation
  - Ogni istanza ha il suo livello di accesso
- Cons:
  - Ogni istanza ha un costo



# Key organization - Select

- Metodo per filtrare chiavi
- Filtro basato sulle chiavi
  - "abc": chiave abc
  - "a\*": chiave che inizia per a
- Filtro per Label
  - "dev": chiavi con tag dev
- Invocazioni multiple filtrano in OR



# Demo

# Refresh configurazione

- Client scaricano valori e mantengono in memoria
- Client può registrarsi a cambiamenti configurazione
  - Sfrutta una chiave *sentinel*
- Client può usare EventGrid
  - sfrutta webhook



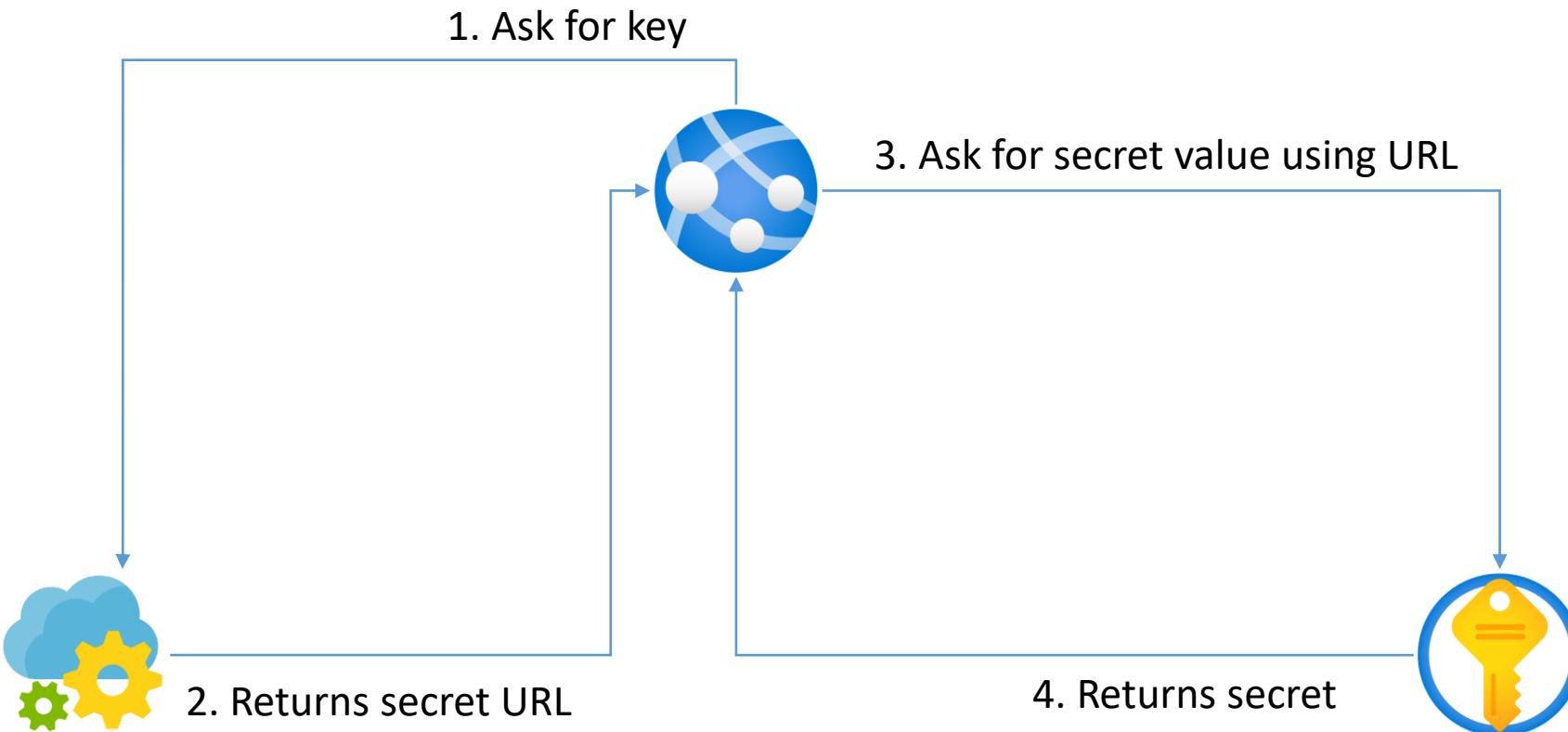
# Demo

# Integrazione con Key Vault

- Chiavi possono puntare a Key Vault
  - Valore è un url a secret in Key Vault
- App configuration non accede a Key Vault
- A runtime SDK recupera valore da Key Vault usando url chiave
  - ConfigureKeyVault
- App Configuration può cripare dati usando una chiave in Key Vault
  - Solo piano standard



# Integrazione con Key Vault



# Demo

# Pricing

	<b>Free</b>	<b>Standard</b>
Instances per subscription	1	Unlimited
Storage per instance	10 MB	1 GB
Revision history	7 days	30 days
Requests Quota	1,000 per day	20,000 per hour
SLA	None	99.9% availability
Security functionality	Encryption with Microsoft-managed keys RBAC support Managed identity	All Free tier functionality plus: Encryption with customer-managed keys Private Link support
Cost	Free	€1.012 per day, plus an overage charge at €0.051 per day The first 200,000 requests are included in the daily charge. Additional requests will be billed as overage.

# Grazie!

- Il materiale sarà online nei prossimi giorni su  
<http://www.communitydays.it>