Evaluando Ambientes de Desarrollo

Grethel Bello Cagnant

grethel.bello@wizeline.com

www.wizeline.com

Acerca de mí

Grethel Bello Cagnant

- Staff Software Engineer
- +9 años de experiencia
- 4 años en Wizeline

proprietary + confidential

www.wizeline.com

Puntos importantes

Identifícate en Zoom utilizando tu nombre y apellido.

Mantén tu micrófono apagado durante el transcurso de la sesión.

Utiliza el chat para hacer tus preguntas durante la sección de Q&A.

Procura enfocar tus preguntas al tema presentado.

Apaga tu cámara en caso de tener problemas con tu conexión.

proprietary + confidential

www.wizeline.com

Código de conducta

Sé respetuoso, no hay malas preguntas o ideas.

Sé cordial y paciente.

Sé cuidadoso con tus palabras.

proprietary + confidential

www.wizeline.com

Objetivo

Al final de esta sesión podrás:

- Examinar los controles de seguridad y las prácticas de un ambiente de desarrollo

proprietary + confidential

www.wizeline.com

Tabla de Contenidos

proprietary + confidential

www.wizeline.com

Zero Trust Architecture

Confia pero verifica

proprietary + confidential

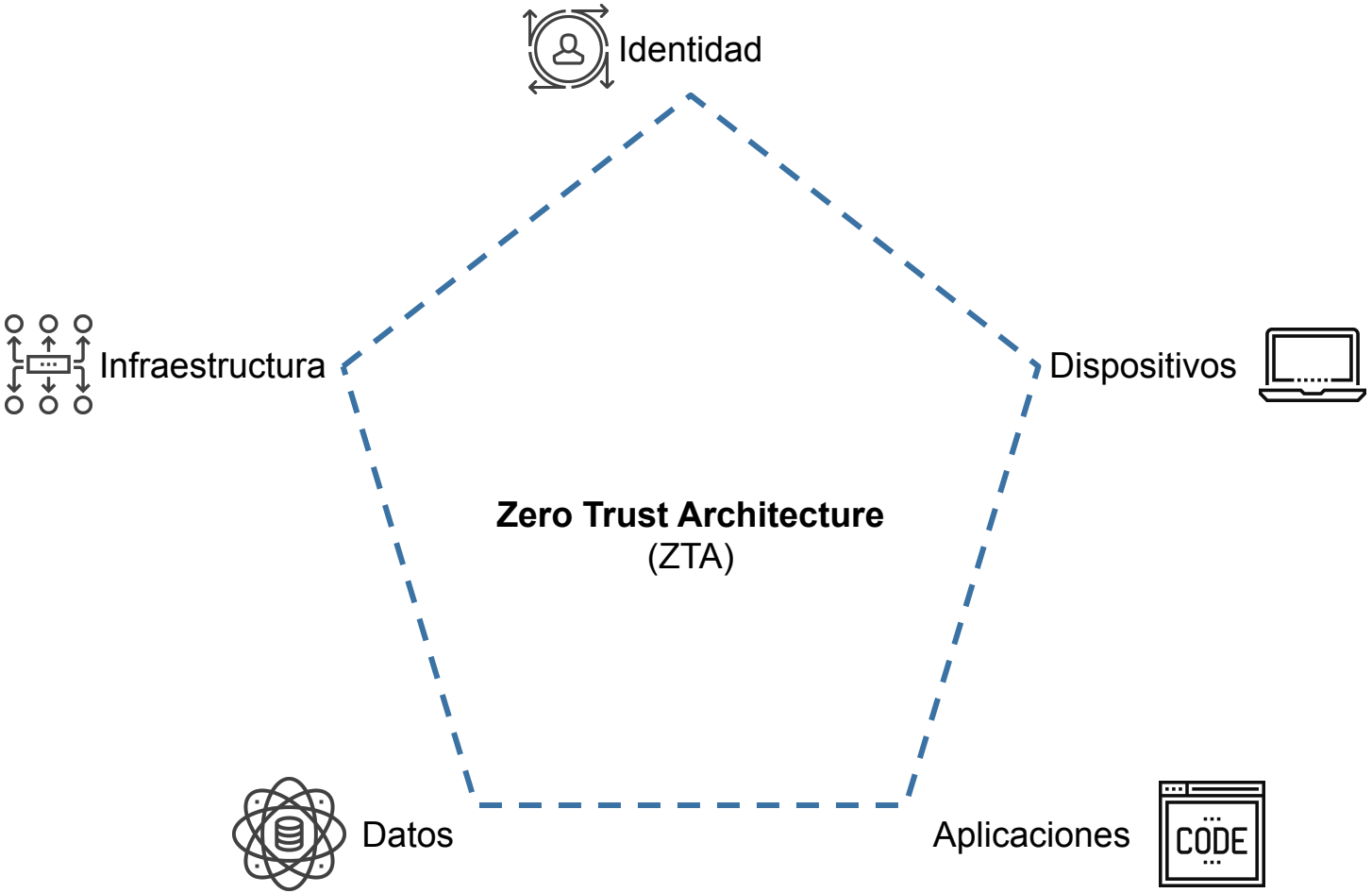
www.wizeline.com

USD $4.24 M

Es el costo promedio de las empresas cuando tienen una brecha de datos.
Fuente: IBM Report: Cost of a Data Breach Hits Record High During Pandemic

proprietary + confidential

www.wizeline.com



proprietary + confidential

www.wizeline.com

Identidad

- Accesos solo en el momento y solo lo que necesita (Just-In-Time & Just-Enough)
- Unificación de autenticación
  - Autenticación de dos pasos
  - Políticas de contraseñas seguras
- Tokens de accesos
- Acceso Remoto: SSH en lugar de SSL

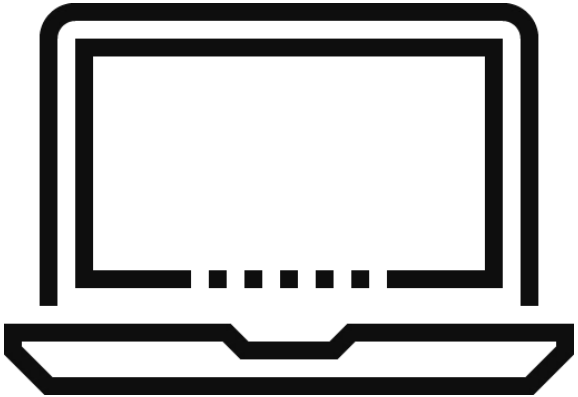
proprietary + confidential

www.wizeline.com

Dispositivos

- Políticas de salud y cumplimiento de requisitos de los dispositivos
- Políticas para restringir conexiones de dispositivos extraíbles
- Controles de aplicación
- VPN
- Monitoreo
- Búsqueda de dispositivos
- Detección y respuesta de dispositivos
- Mecanismos de encriptación
- Análisis de Riesgo

National Cyber Security Centre

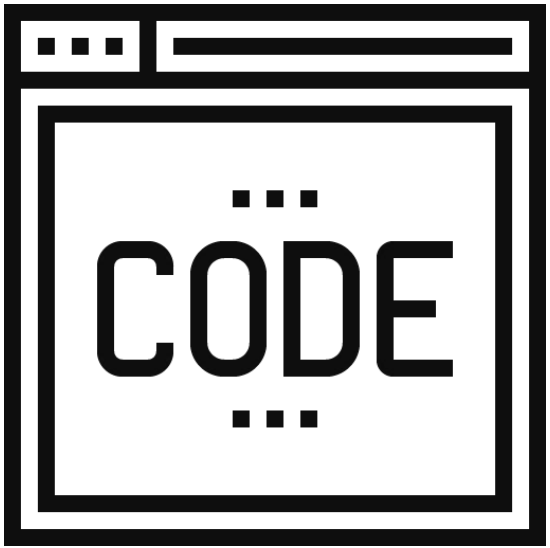


proprietary + confidential

www.wizeline.com

Aplicaciones

- Usar secretos en lugar de contraseñas en código en texto plano
- Evitar mostrar información sensible en los logs
- Software Composition Analysis
- Blame Code
- Restricciones de merge y revisores encargados



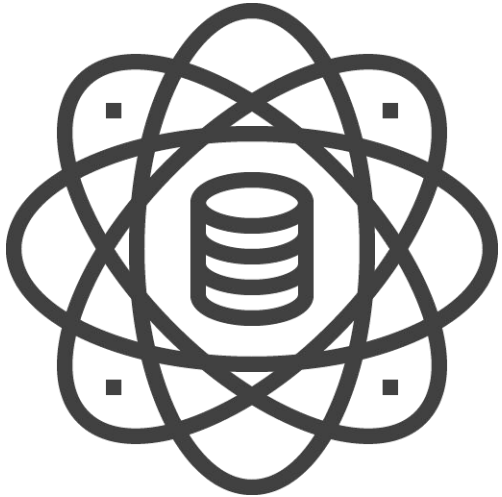

```
$ git config --global user.name "John Doe"
$ git config --global user.email johndoe@example.com
```

proprietary + confidential

www.wizeline.com

Datos

- Clasificación y Etiquetado
- Protección de la información
  - Acceso a datos sensibles a través de los protocolos de control
  - Administración de derechos
  - Encripción
- Prevención de pérdida de datos
- Administración de riesgo interno
  - Investigación
  - Plan
  - Políticas
  - Implementación
- Gobierno de datos

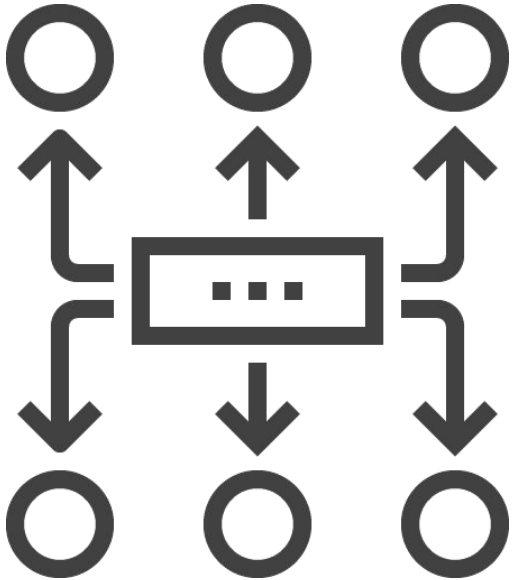


proprietary + confidential

www.wizeline.com

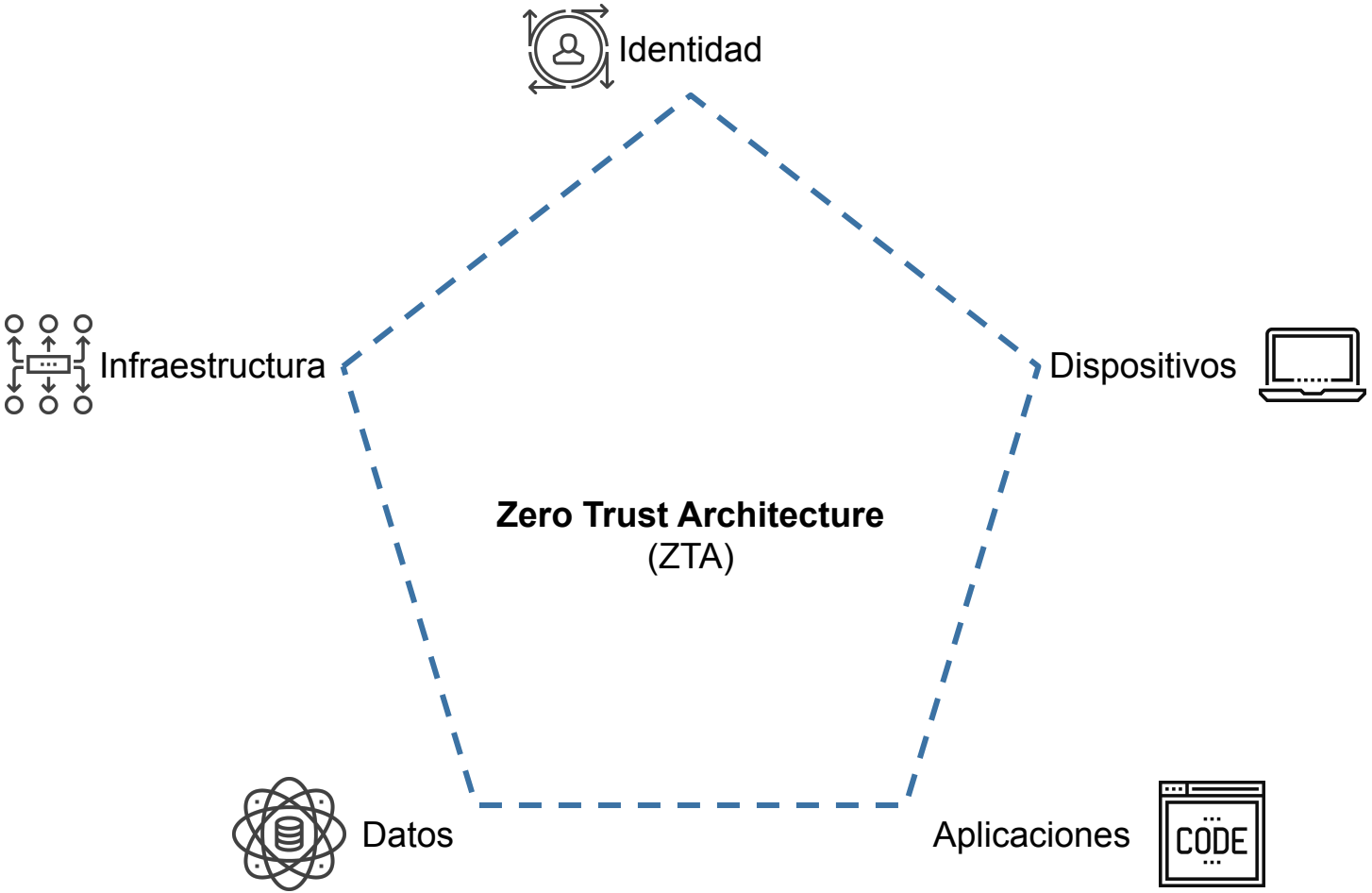
Infraestructura

- Infraestructura como Código
- Versionamento
- Administración de configuraciones
- JIT/JEA para la administración de privilegios
- Usar telemetría para detectar ataques y anomalías
- Automáticamente bloquear e identificar comportamientos riesgosos y tomar acciones de protección



proprietary + confidential

www.wizeline.com



proprietary + confidential

www.wizeline.com

Tabla de Contenidos

proprietary + confidential

Case of Study

www.wizeline.com

Caso de Estudio

La factura mensual del proveedor de la nube llegó más elevada de lo usual. Después de analizar el reporte de costos, el incremento se debió a un servidor consumiendo más CPU de lo normal.

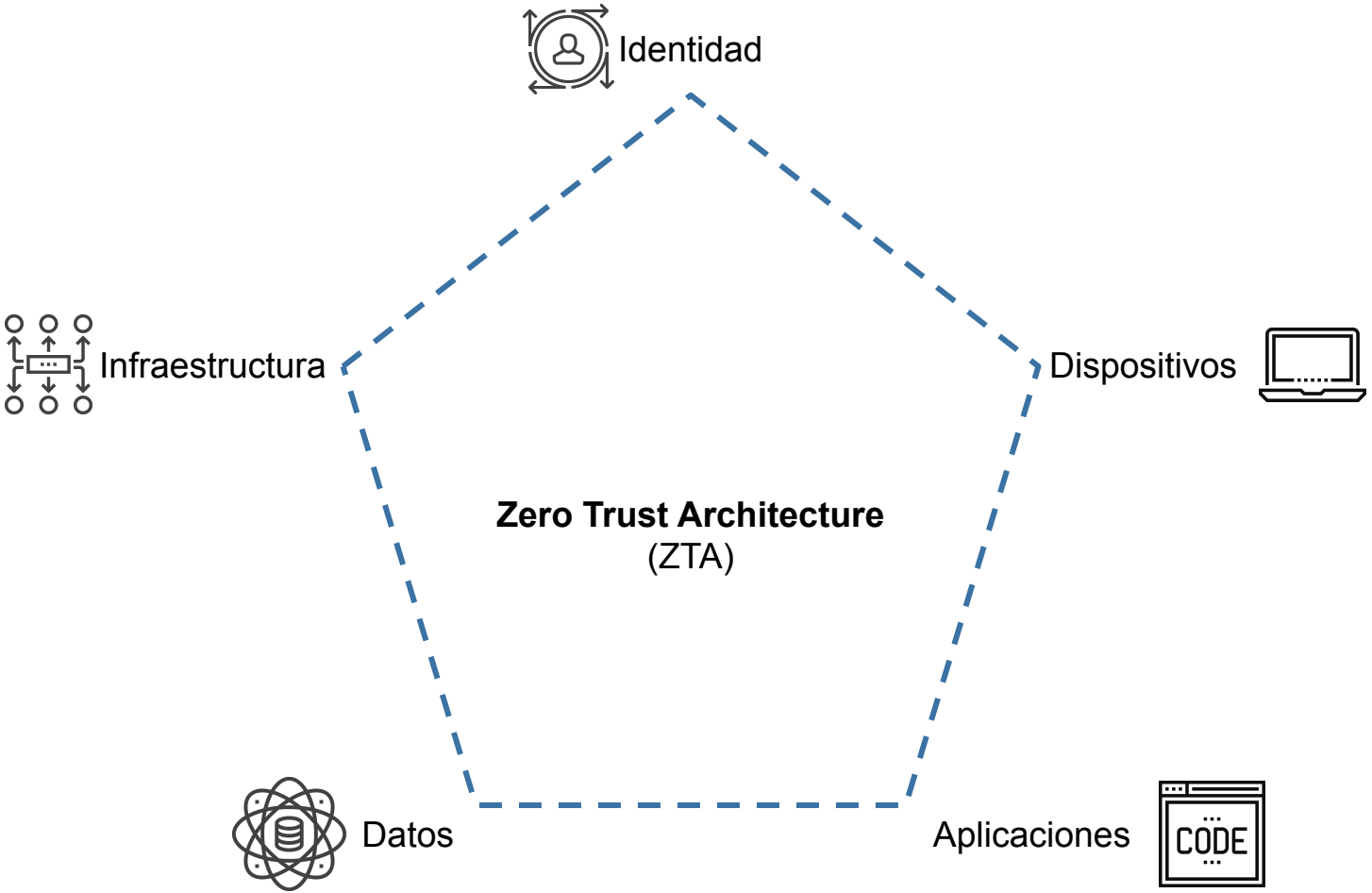
El servidor contenía Jenkins, pero su histórico de actividad no demostraba un incremento en su uso. Al revisar los procesos ejecutándose en el servidor, se descubrió un script de cripto minería.

La vulnerabilidad se debió a que el servidor interno estaba expuesto a Internet y Jenkins no estaba configurado para requerir autenticación. Además, las entradas y salidas de la red estaban abiertas a todo el tráfico en todos los puertos, por lo tanto el script no tuvo problema para comunicarse.

Alguien estaba minando cripto con los recursos de la empresa.

proprietary + confidential

www.wizeline.com



proprietary + confidential

www.wizeline.com

Caso de Estudio

La alarma de escritura/lectura por minuto de la base de datos empezó a activarse hace dos semanas. Se logró identificar el sistema generando más escrituras de lo usual por los queries en los logs.

Después de revisar el código, lo cual tomó 5 días, se encontraron fragmentos que no pertenecen a la lógica de negocio. No se puede identificar a la persona que creó el código ya que los usuarios del repositorio remoto no están relacionados con los usuarios de la organización. Sin embargo se identificó al usuario que creó este usuario desconocido.

La auditoría determinó que debido a una brecha de usuario, se creó un usuario fantasma el cual agregó código malicioso al repositorio. Debido a la falta de restricciones de permisos y una SSO para la autenticación se tardó un total de 7 semanas en identificar la vulnerabilidad.

proprietary + confidential

www.wizeline.com

Tabla de Contenidos

proprietary + confidential

www.wizeline.com

Resumen

Recordemos que en esta sesión hablamos de:

- Entender como funciona el Zero Trust Architecture
- Análisis de caso de estudio

proprietary + confidential

www.wizeline.com

Objetivo

Al final de esta sesión podrás:

- Examinar los controles de seguridad y las prácticas de un ambiente de desarrollo

proprietary + confidential

¿Preguntas?

www.wizeline.com

proprietary + confidential

www.wizeline.com

Feedback Form

Was this template helpful?
Let us know your feedback

proprietary + confidential

www.wizeline.com

Gracias.

proprietary + confidential