



Modelado de Amenazas

Una Introducción

Hernán M. Racciatti, CISSP, CSSLP, CEH

•

SIClabs

hracciatti@siclabs.com

@my4ng3l



Agenda

- Algunas Definiciones
- Modelado de Amenazas
 - Qué?
 - Para qué?
 - Por qué?
 - Cuando? (SDLC)
 - Cómo? (El Proceso)
- Conclusiones
- Referencias y Lectura Complementaria





Algunas Definiciones

- **Vulnerabilidad**
 - Ausencia o debilidad de un control
- **Amenaza**
 - Evento cuya ocurrencia podría impactar en forma negativa en la organización (Las amenazas *explotan o toman ventaja* de las vulnerabilidades).
- **Riesgo**
 - Combinación de probabilidad de ocurrencia e impacto de una amenaza.



Algunas Definiciones (Cont.)

- **Exposición**

- Instancia en la cual la información o activo de información es susceptible a dañarse o perderse por la explotación de una vulnerabilidad.

- **Contramedida (Salvaguarda o Control)**

- Cualquier tipo de medida que minimice el riesgo asociado con la ocurrencia de una amenaza específica.

- **Exploit**

- Método o programa utilizado para aprovecharse de una vulnerabilidad.



Modelado de Amenazas

Qué?

- Enfoque / Método de Análisis basado en la seguridad.
- Proceso para la evaluación y documentación de los riesgos de seguridad de un sistema.
- Parte crucial de la etapa de diseño.
- Herramienta indispensable a la hora de trabajar sobre la seguridad de las aplicaciones.





Modelado de Amenazas

Qué?

Enfoque

Método

Análisis

Riesgos

Crucial

Diseño

Herramientas

Aplicación





Modelado de Amenazas

Modelo de Amenazas?

Representación de:

- La Superficie de Ataque del Sistema
- Amenazas que puedan afectar el sistema
- Activos que puedan ser comprometidos por la amenaza (El agente de la amenaza).





Modelado de Amenazas

Para qué?

- Comprender el perfil de amenazas a las que está expuesto un sistema.
- Colaborar con el entendimiento respecto de:
 - *Donde el producto es mas vulnerable*
 - *Cuales amenazas requieren ser mitigadas*
 - *Como direccionar las mismas*
- Identificar estrategias de mitigación.
- Justificar la implementación de controles!





Modelado de Amenazas



Quién?

Qué?

Cómo?

Impacto

Mitigación



Modelado de Amenazas

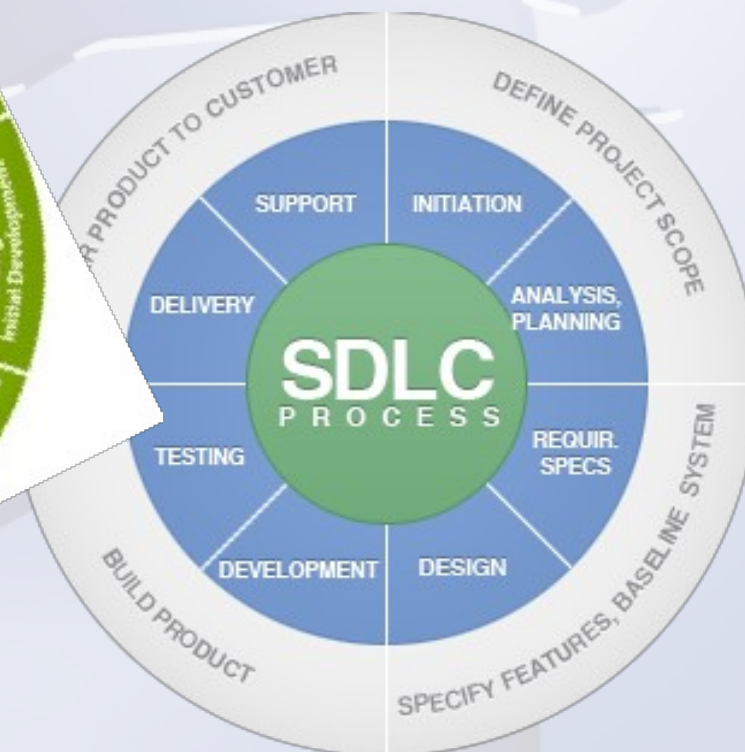
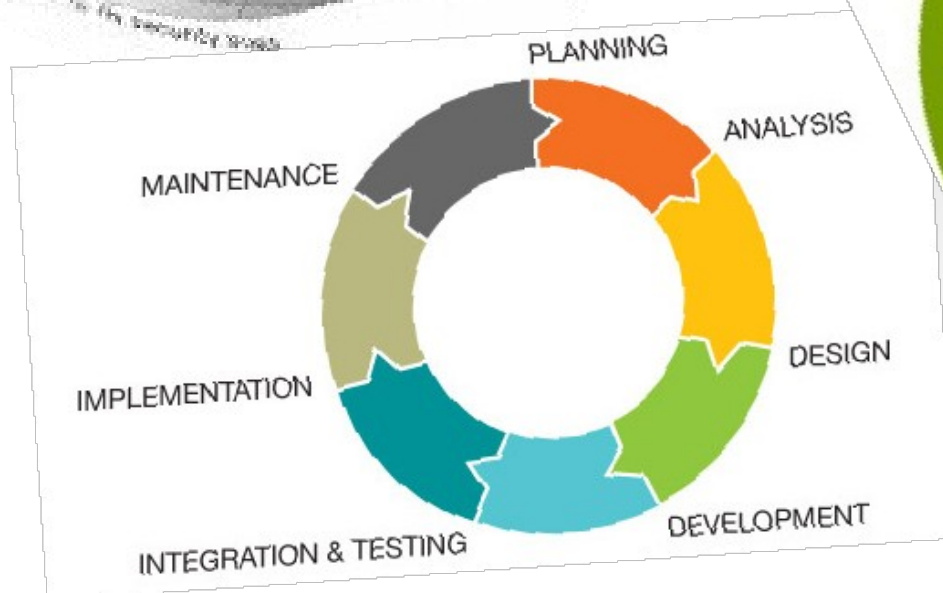
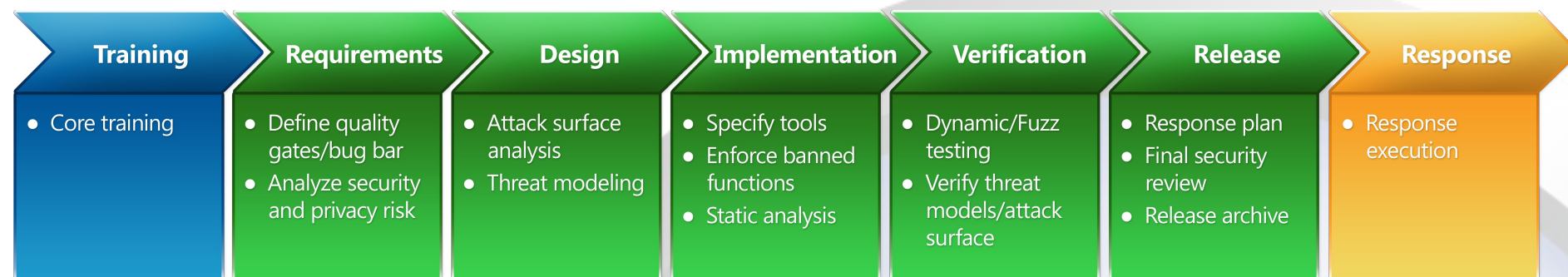
Por qué?

- Colabora con la identificación de complejos bugs de diseño.
- Colabora con el descubrimiento de vulnerabilidades.
- Colabora con el proceso general de reducción del riesgo.
- Complementa las especificaciones del diseño de seguridad.
- Colabora en la integración de nuevos miembros al equipo de desarrollo.
- Reduce el costo de asegurar una aplicación.
- Soporta aplicaciones seguras por diseño.
- Provee un proceso.



SDLC

Cuando?





SDLC (Cont.)

Modelado de

Revisión de Código

Penetration Test

Codificación

Diseño

Testing/Validación



SDLC (Cont.)

Modelado de

Revisión de Código

Penetration Test

Codificación

Diseño

Testing/Validación



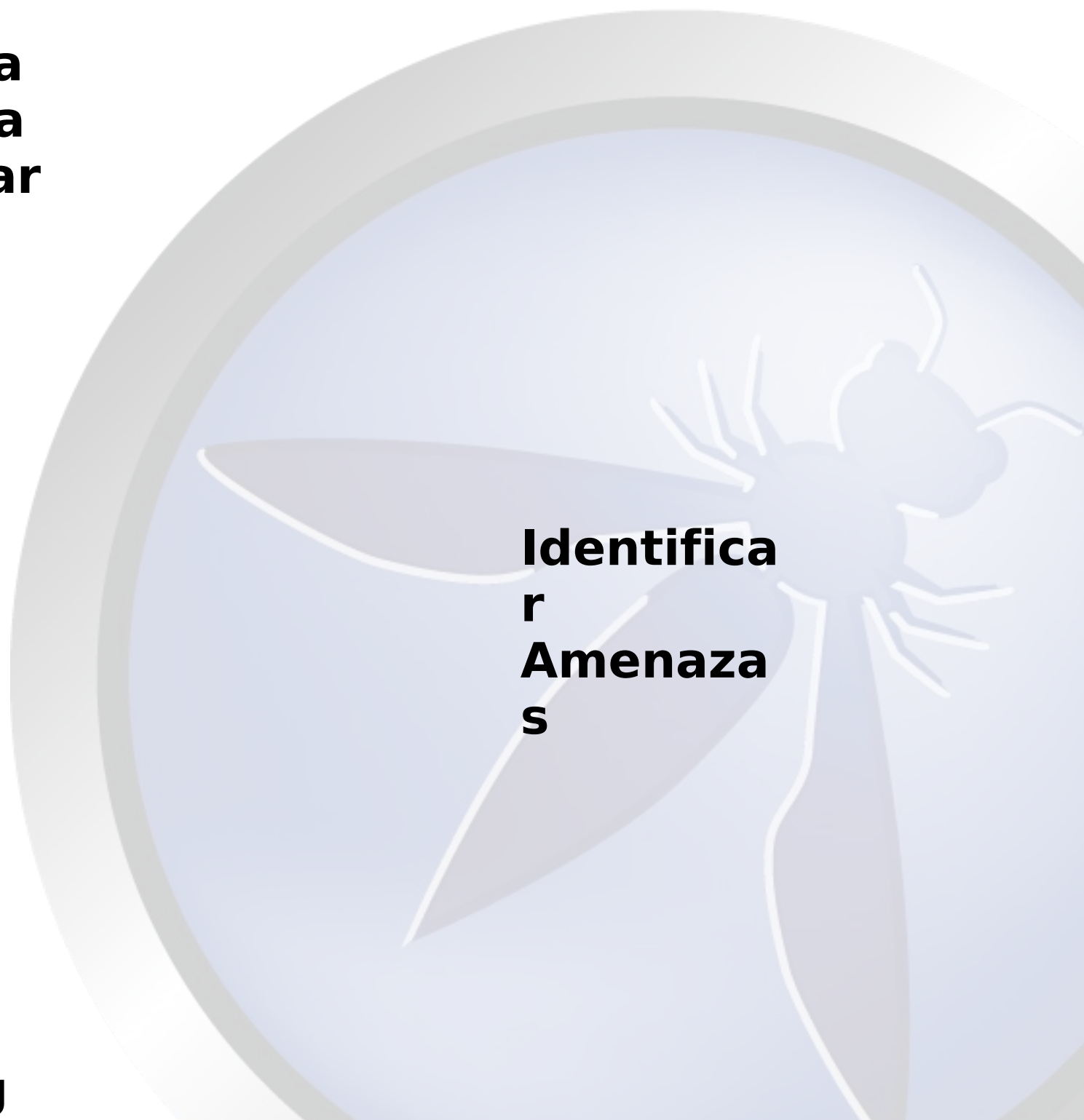
Cómo? (El Proceso)

**Dia
gra
mar**

**Va
lid
ar**

**Identifica
r
Amenaza
s**

**Mi
tig
ar**





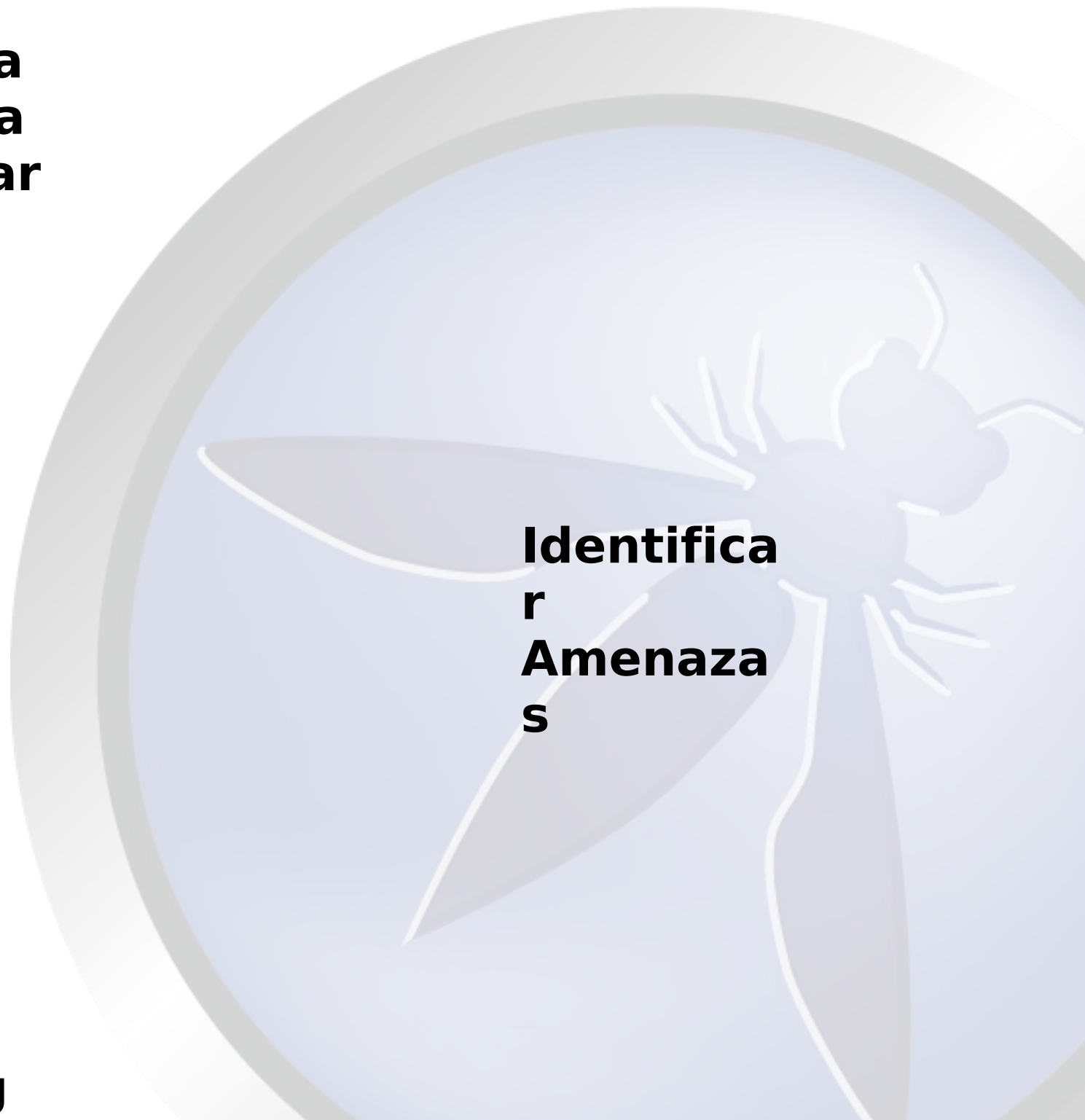
Diagramar

**Dia
gra
mar**

**Va
lid
ar**

**Identifica
r
Amenaza
s**

**Mi
tig
ar**





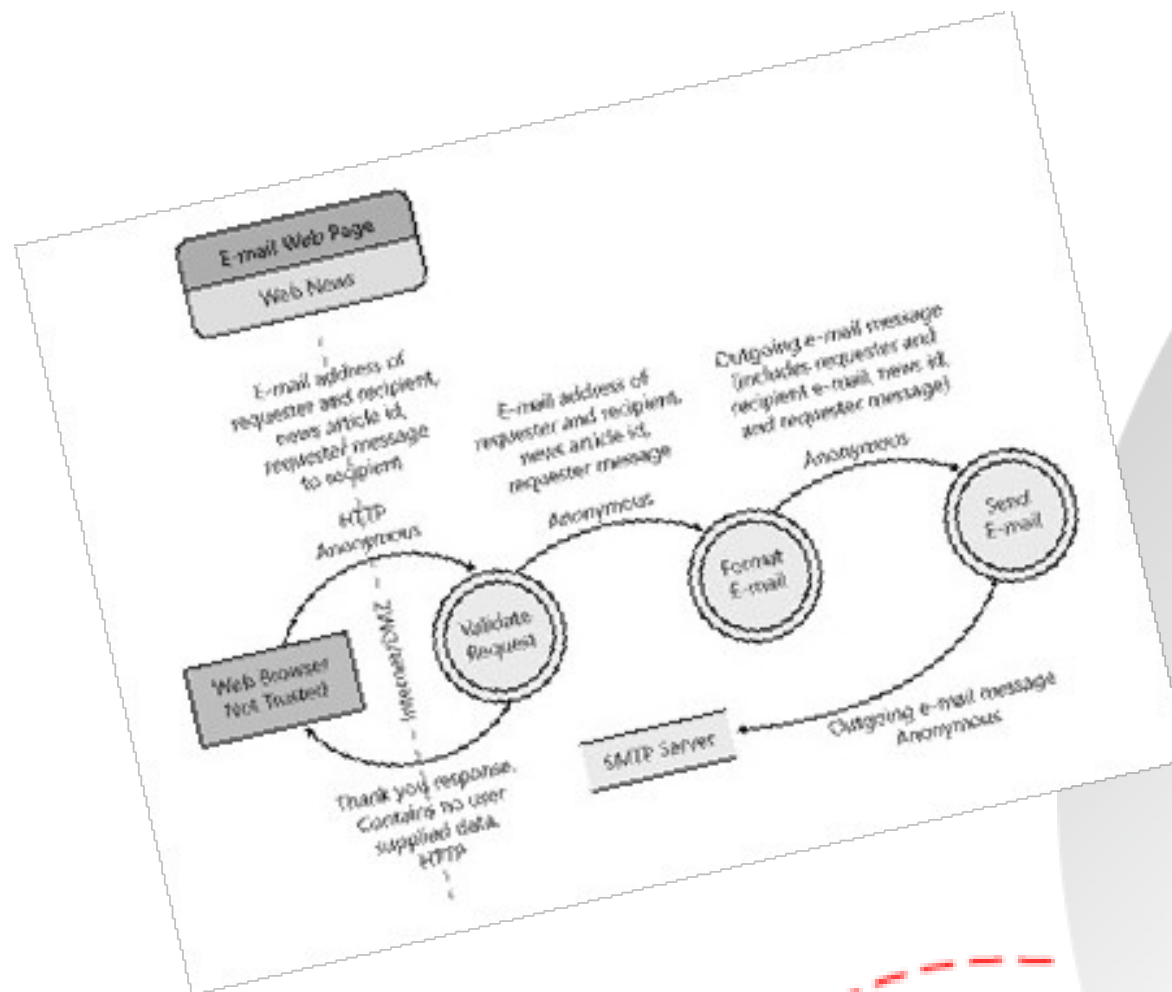
Diagrammar (Cont.)





Diagrammar (Cont.)

Elementos



Trust
Boundary

Data Flow

Data Store

External
Entity

Proces
S

Multiple
Process



Diagramar (Cont.)

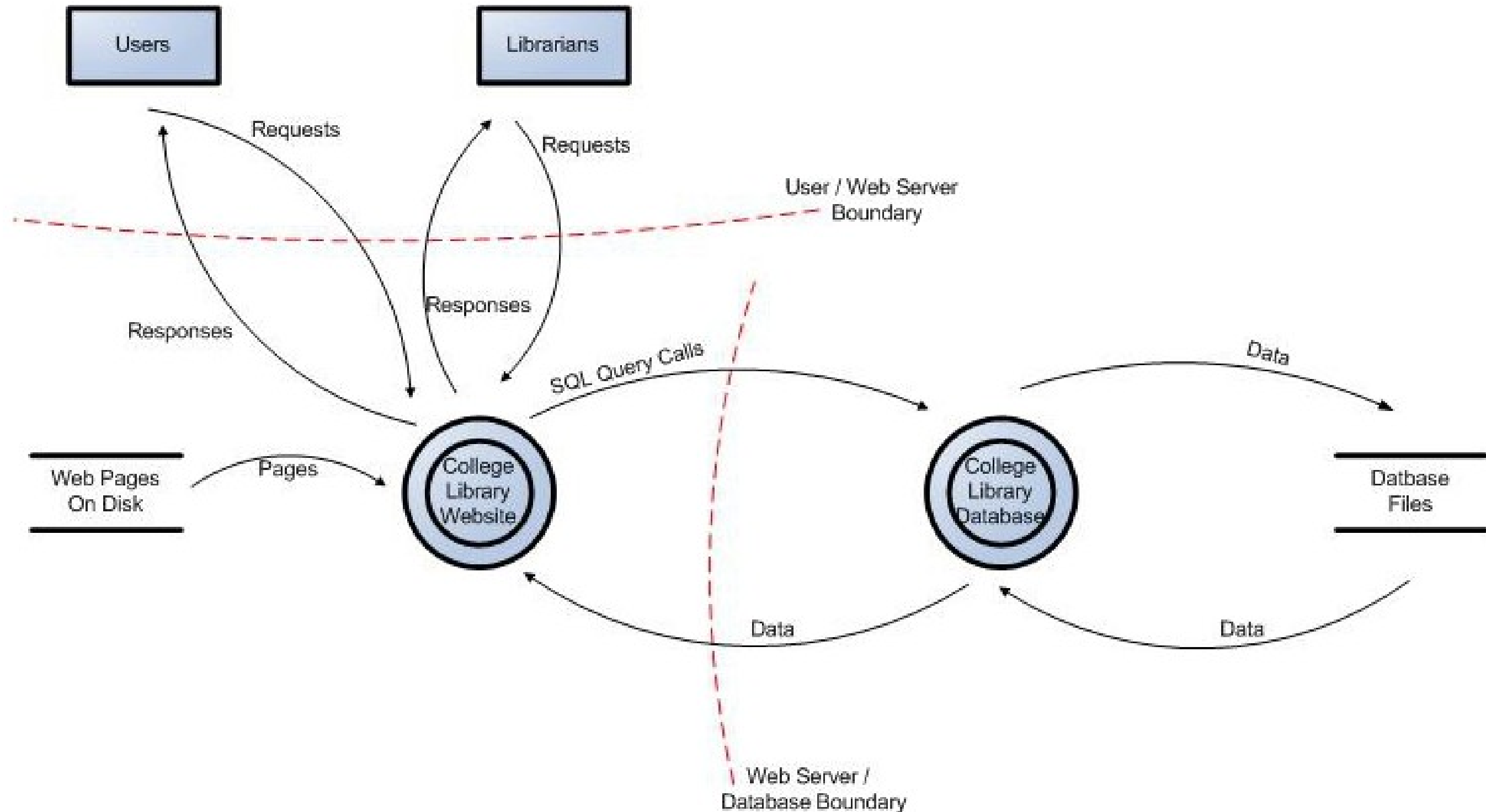
Diferentes Niveles

- Diagrama de Contexto
 - Muy alto nivel (El producto/sistema entero)
- Nivel 1
 - Alto nivel (Una unica funcionalidad o un escenario)
- Nivel 2
 - Bajo nivel (Subcomponente o función detallada)
- Nivel 3
 - Mas detallado



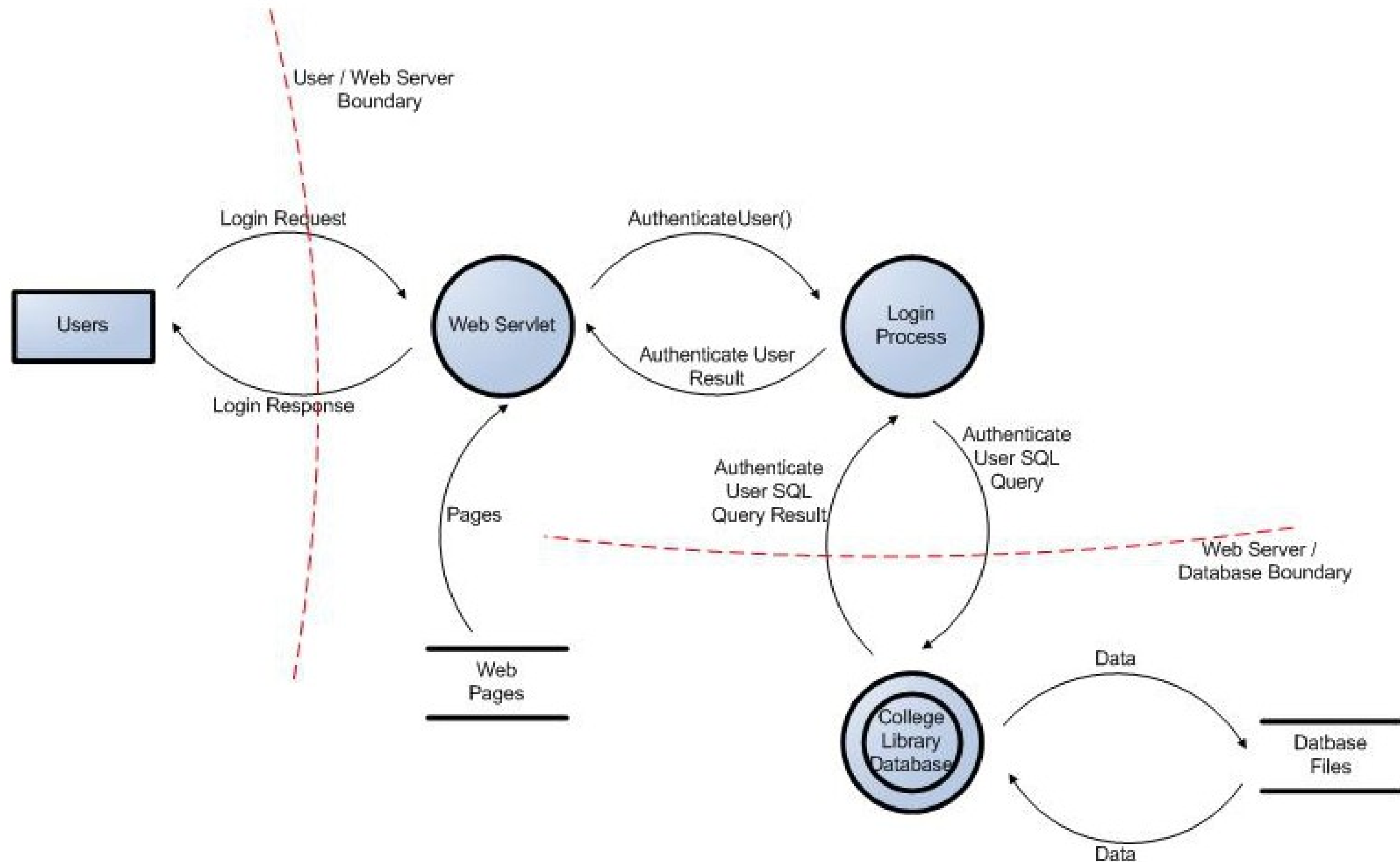


Diagrammar (Cont.)





Diagrammar (Cont.)



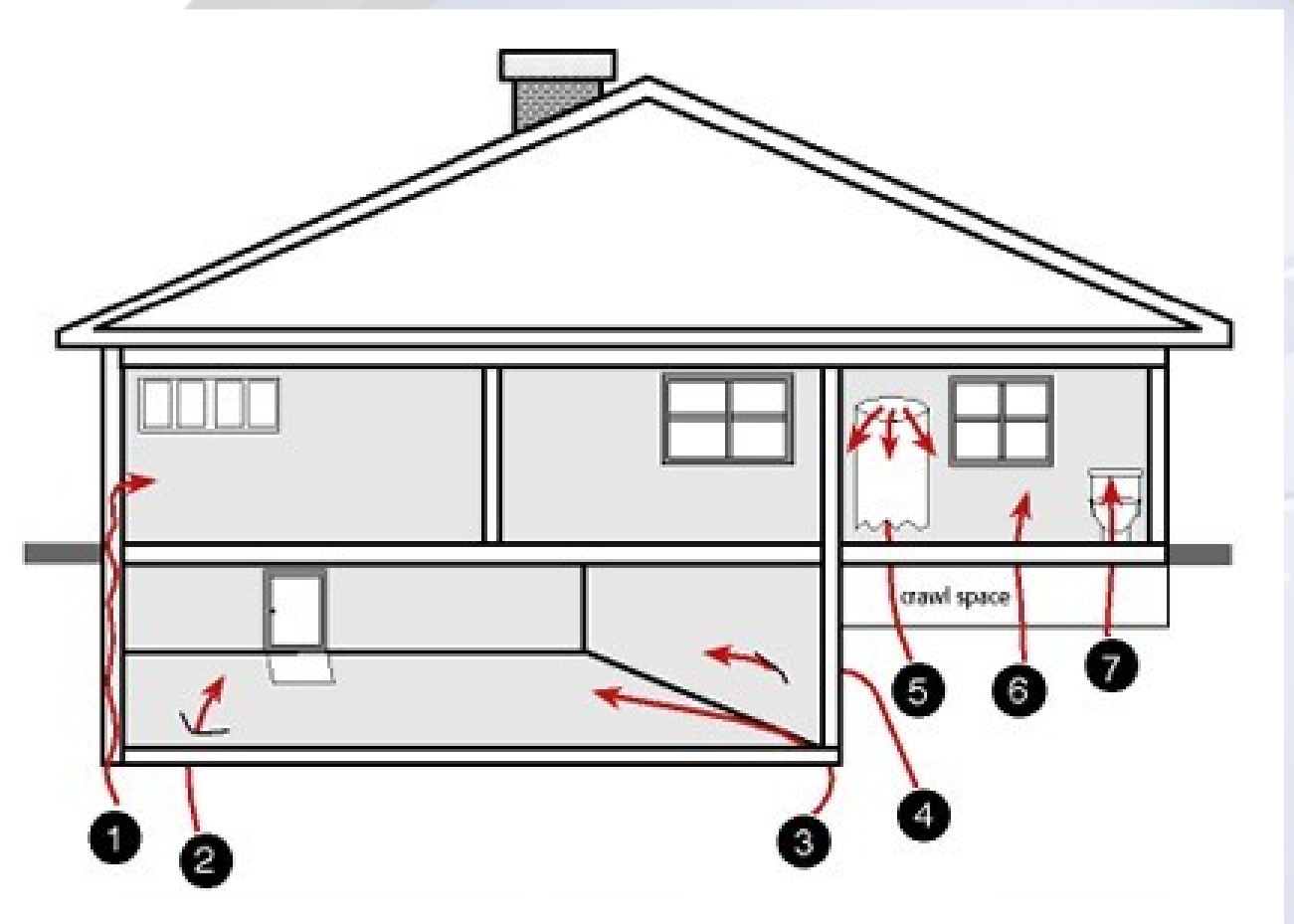


Diagrammar (Cont.)

Puntos de Entrada

- User Interface
- Files
- Sockets
- HTTP Requests
- Named Pipes
- APIs
- Registry
- E-mail
- Command Arguments
- Environment Variables
- Etc.

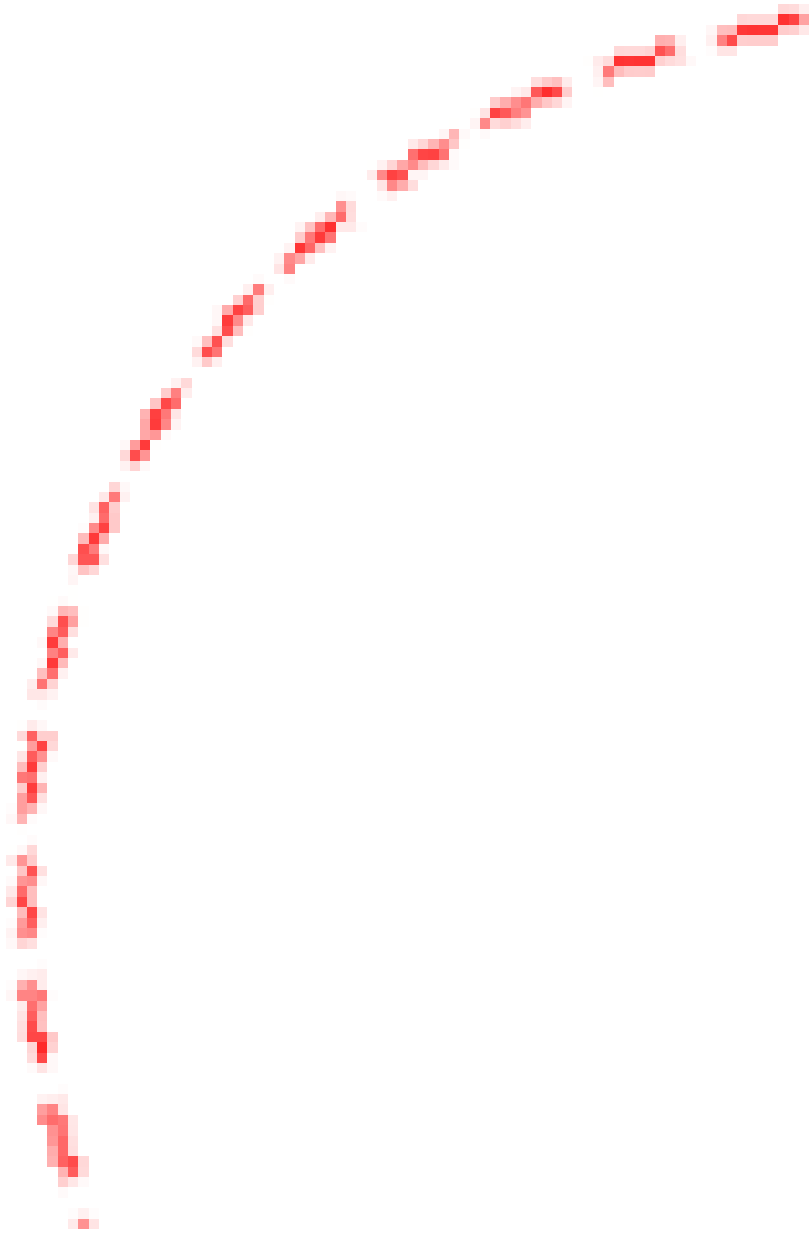
Line





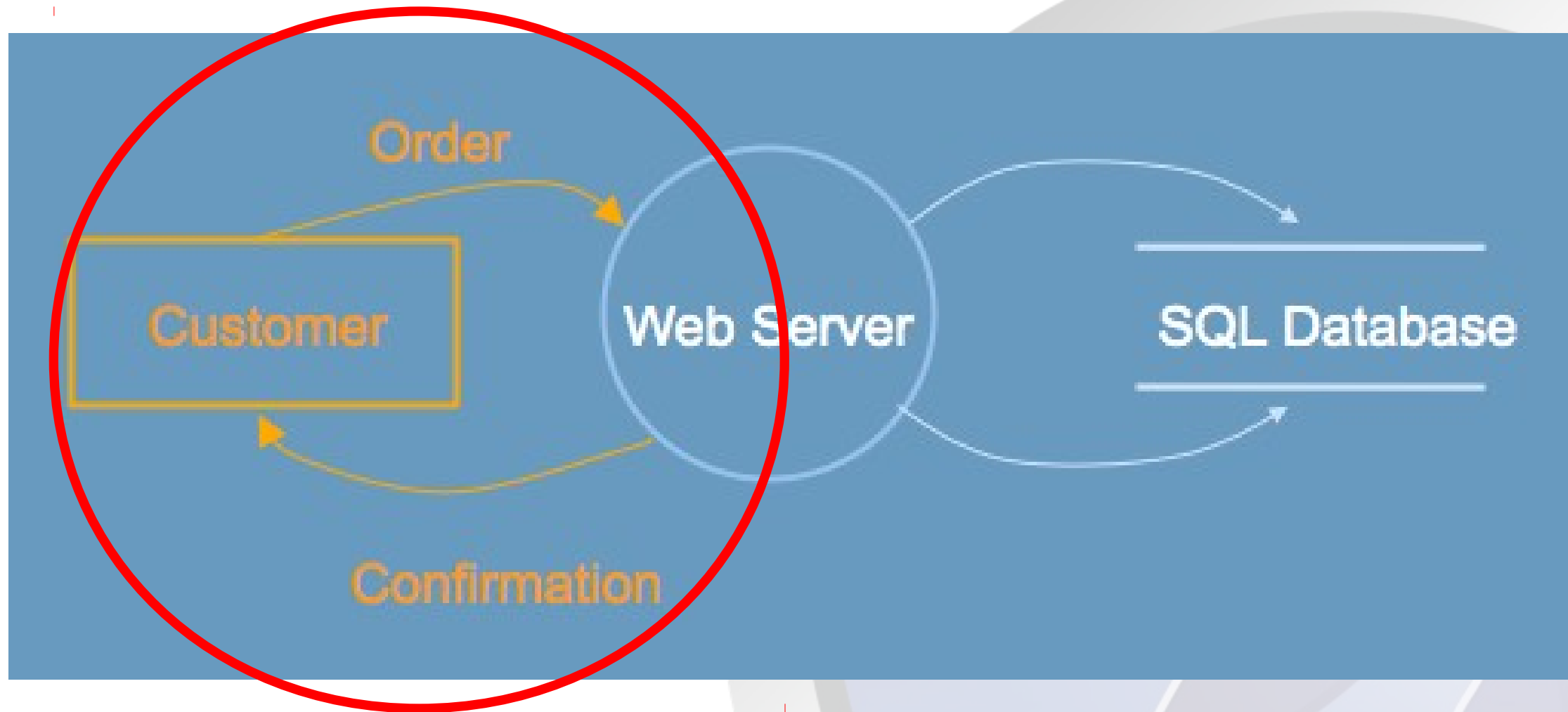
Diagrammar (Cont.)

Trust Boundary





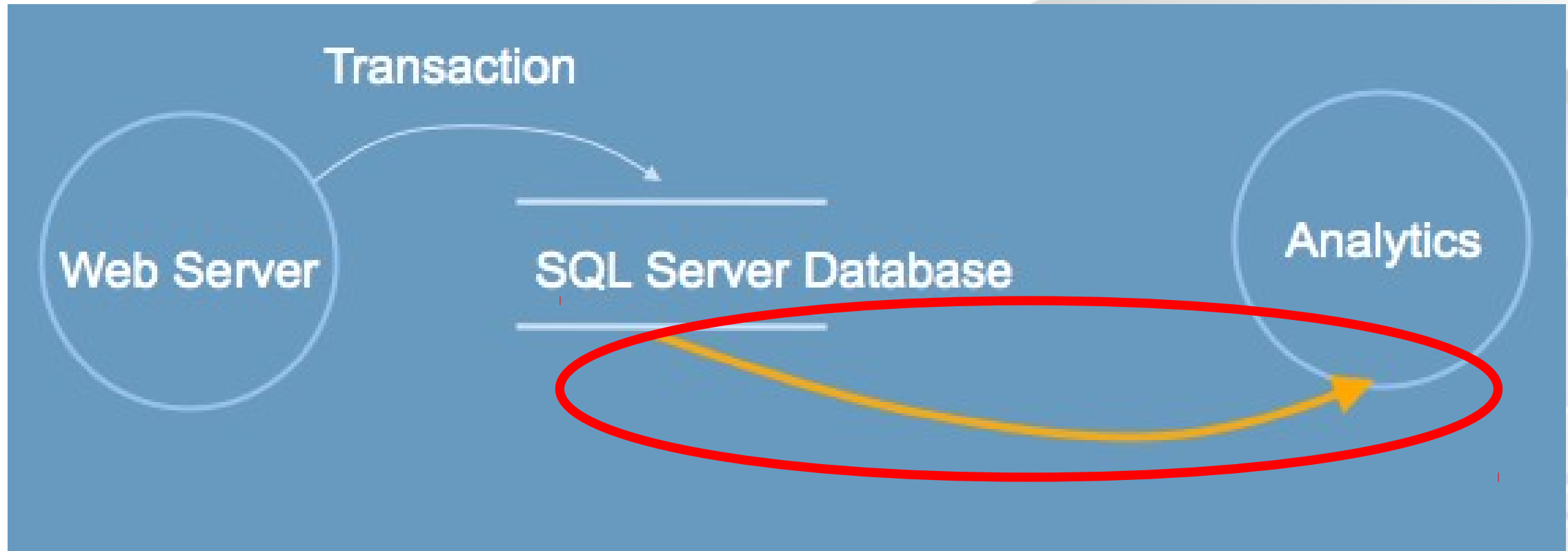
Diagramar (Cont.)



*“Los datos no aparecen magicamente...
provienen de entidades externas o de data
stores”*



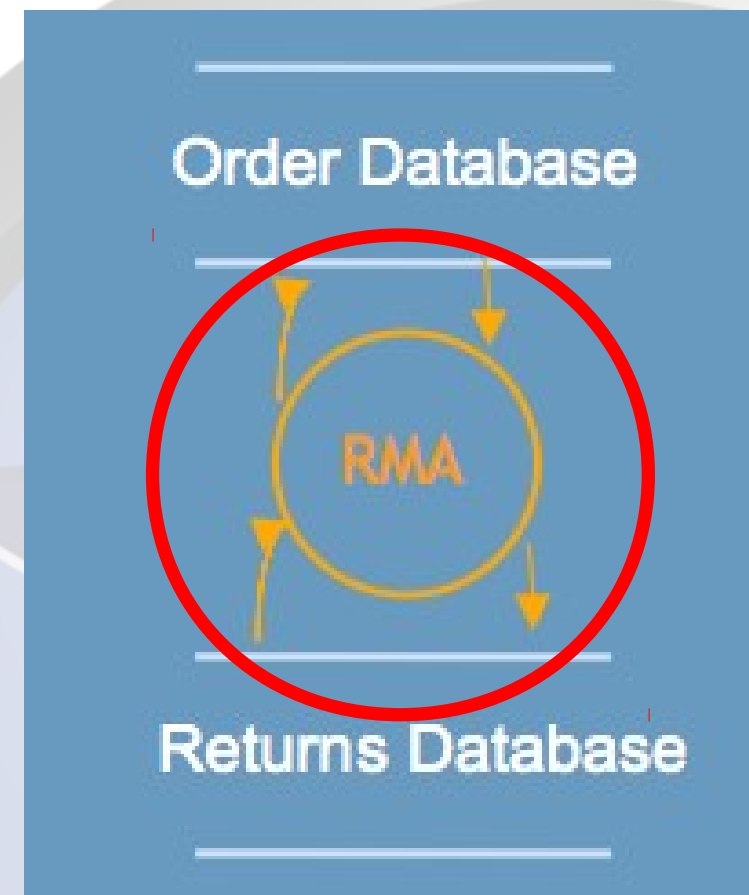
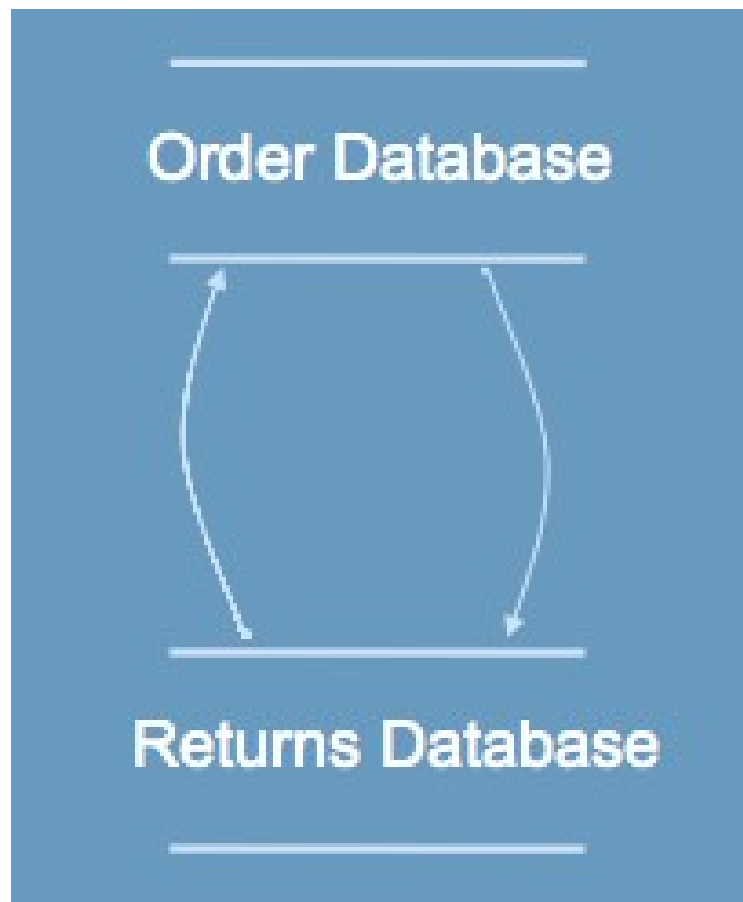
Diagramar (Cont.)



“Existen los contenedores de datos... claro y alguien seguramente los usa”



Diagramar (Cont.)



“Los datos no fluyen magicamente... estos lo hacen a travéz de procesos...”



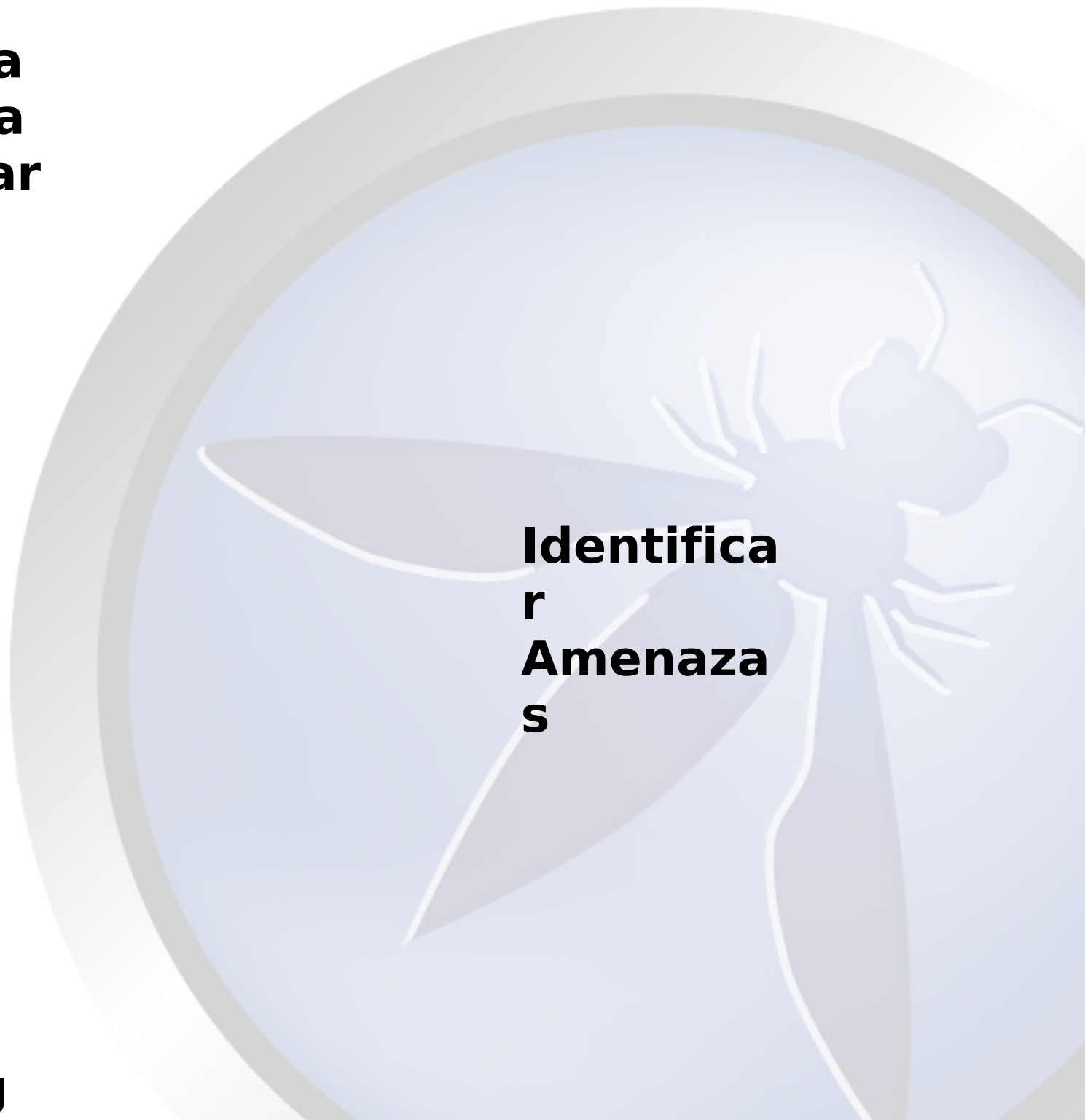
Identificar Amenazas

**Dia
gra
mar**

**Va
lid
ar**

**Identifica
r
Amenaza
s**

**Mi
tig
ar**





Identificar Amenazas

(Cont.)

S Spoofing (Se puede acceder con una identidad falsa?)

T Tampering (Se pueden modificar datos de modo no autorizado?)

R Repudiation (Puede un atacante repudiar sus acciones?)

I Information Disclosure (Se puede acceder a información reservada?)

D Denial of Service (Es posible disminuir la dispo. de una aplicación?)

E Elevation of Privilege (Puede un atacante elevar sus privilegios?)



Identificar Amenazas (Cont.)

- Amenaza: **S**poofing
- Propiedad: Autenticación
- Definición: Impersonar alguien o algo.
- Ejemplo: Pretender ser un amigo de wanda nara, owasp.org o ntdll.dll

STRIDE



Identificar Amenazas (Cont.)

- Amenaza: **T**ampering
- Propiedad: Integridad
- Definición: Modificar datos o código
- Ejemplo: Modificar una DLL sobre el HD o un paquete atravesando la red

STRIDE



Identificar Amenazas (Cont.)

- Amenaza: **R**epudiation
- Propiedad: No-Repudio
- Definición: Prertender no haber realizado una acción
- Ejemplo: “Yo no envíe ese mail”, “Yo no modifique ese archivo”, “Yo no visite ese sitio web!”

STRIDE



Identificar Amenazas (Cont.)

- Amenaza: **I**nformation Disclosure
- Propiedad: Confidencialidad
- Definición: Exponer información a alguien no autorizado a ver esta
- Ejemplo: Permitir a alguien leer el código fuente de mi aplicación, publicar una lista de clientes en la web

STR**I**DE



Identificar Amenazas (Cont.)

- Amenaza: **D**enial of Service
- Propiedad: Disponibilidad
- Definición: Denegar o degradar el servicio a los usuarios
- Ejemplo: Hacer caer un servicio o un sitio web, enviando paquetes malformados, absorbiendo segundos de CPU, etc.

STRIDE




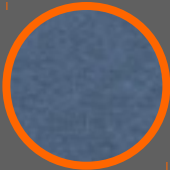

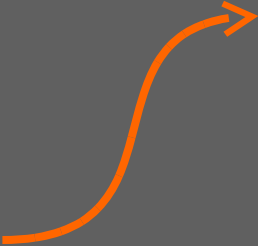
Identificar Amenazas (Cont.)

- Amenaza: **E**levación de Privilegios
- Propiedad: Autorización
- Definición: Ganar capacidades sin la debida autorización
- Ejemplo: Usuario limitado, ganando privilegios de admin

STRIDE**E**



Identificar Amenazas

Elemento	S	T	R	I	D
	<input type="checkbox"/>		<input type="checkbox"/>	E	
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>	?	<input type="checkbox"/>	<input type="checkbox"/>
		<input type="checkbox"/>		<input type="checkbox"/>	<input type="checkbox"/>



Analizar Amenazas

D **Damage** (Cuan daño puede causar?)

R **Reproducibility** (Cuan dificultosa es de reproducir?)

E **Exploitability** (Qué tan fácil es de explotar?)

A **Affected Users** (Qué cantidad de usuarios pueden verse afectados?)

D **Discoverability** (Que tan fácil es su descubrimiento?)



Analizar Amenazas (Cont.)

	Alto (3)	Medio (2)	Bajo (1)
Damage	El agresor puede obtener datos muy sensibles, dañar servidores etc.	Puede obtener datos sensibles	Puede acceder a datos poco importantes
Reproducibility	Siempre es posible	Sucede si se realiza en un corto tiempo	Raramente se puede concretar
Exploitability	puede hacerlo	Se deben tener ciertos conocimientos	Tal vez alguno
Affected Users	La mayoría	Algunos	Pocos, si es que hay alguno
Discoverability	Fácil de ver	Más costoso de ver	Muy difícil de encontrar



Analizar Amenazas (Cont.)

Combinando STRIDE con DREAD

- Calculo del Nivel de Riesgo de una Amenaza
- Alto/Medio/Bajo – 1 a 10
- Ejemplo:
 - *Amenaza #1: Usuario malicioso visualiza información confidencial*
 - *STRIDE: Information Disclosure*
 - *Damage: 8*
 - *Reproducibility: 10*
 - *Exploitability: 7*
 - *Affected Users: 10*
 - *Discoverability: 10*
 - *Valor de Riesgo DREAD: $(8+10+7+10+10/5)=9$*



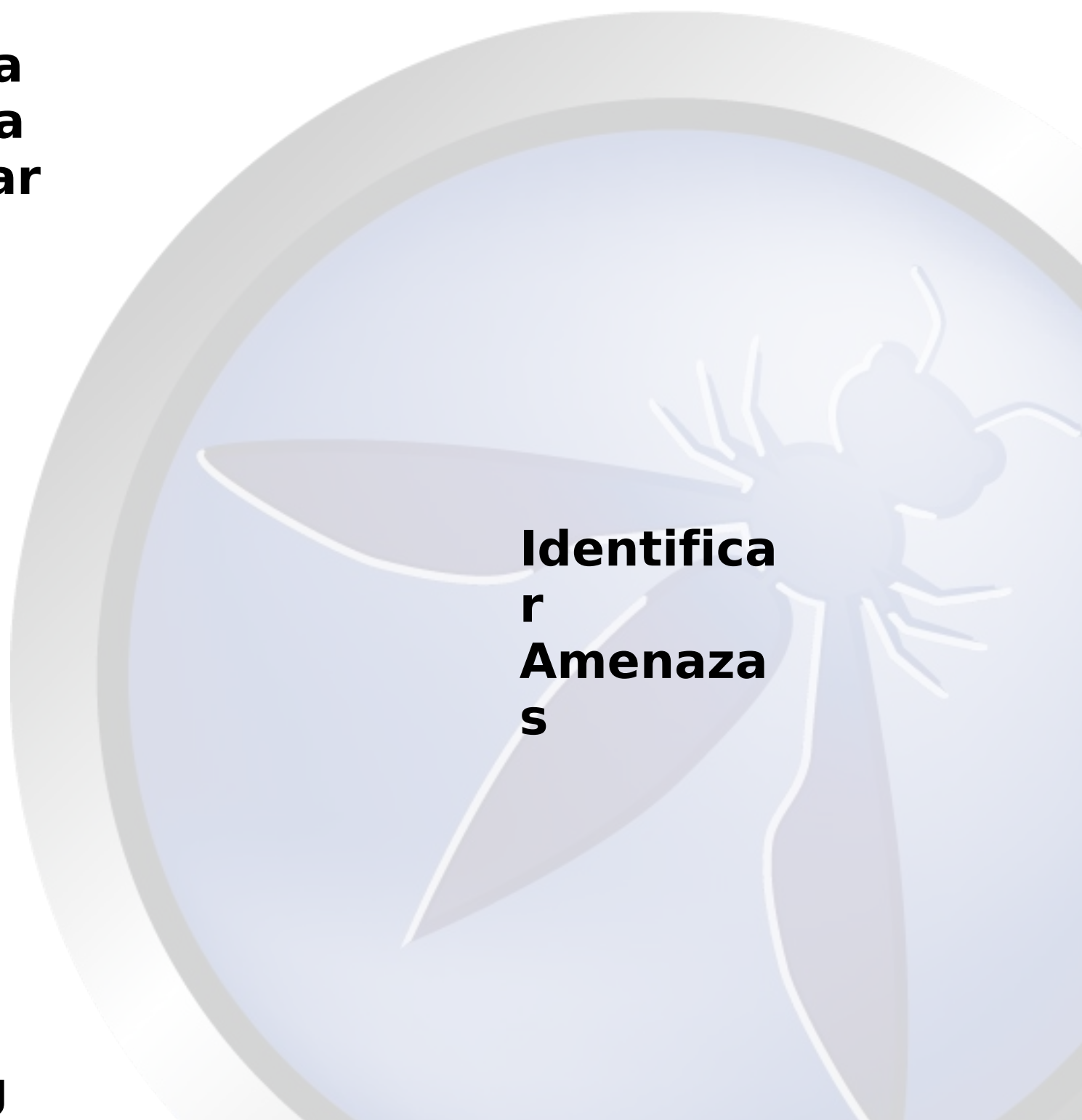
Mitigar

**Dia
gra
mar**

**Va
lid
ar**

**Identifica
r
Amenaza
s**

**Mi
tig
ar**





Mitigar (Cont.)

Authentication:

- Ensure all internal and external connections (user and entity) go through an appropriate and adequate form of authentication. Be assured that this control cannot be bypassed.
- Ensure all pages enforce the requirement for authentication.
- Ensure that whenever authentication credentials or any other sensitive information is passed, only accept the information via the HTTP "POST" method and will not accept it via the HTTP "GET" method.
- Any page deemed by the business or the development team as being outside the scope of authentication should be reviewed in order to assess any possibility of security breach.
- Ensure that authentication credentials do not traverse the wire in clear text form.
- Ensure development/debug backdoors are not present in production code.

Authorization:

- Ensure that there are authorization mechanisms in place.
- Ensure that the application has clearly defined the user types and the rights of said users.
- Ensure there is a least privilege stance in operation.
- Ensure that the Authorization mechanisms work properly, fail securely, and are not bypassed.
- Ensure that authorization is checked on every request.
- Ensure development/debug backdoors are not present in production code.

Cookie Management:

- Ensure that sensitive information is not comprised.
- Ensure that unauthorized activities cannot take place via cookie manipulation.
- Ensure that proper encryption is in use.
- Ensure secure flag is set to prevent accidental transmission over "the wire" in a non-secure manner.
- Determine if all state transitions in the application code properly check for the cookies and enforce their use.
- Ensure the session data is being validated.
- Ensure cookies contain as little private information as possible.
- Ensure entire cookie is encrypted if sensitive data is persisted in the cookie.
- Define all cookies being used by the application, their name, and why they are needed.

Data/Input Validation:

- Ensure that a DV mechanism is present.
- Ensure all input that can (and will) be modified by a malicious user such as HTTP headers, input fields, hidden fields, drop down lists, and other web components are properly validated.
- Ensure that the proper length checks on all input exist.
- Ensure that all fields, cookies, http headers/bodies, and form fields are validated.
- Ensure that the data is well formed and contains only known good chars if possible.
- Ensure that the data validation occurs on the server side.
- Examine where data validation occurs and if a centralized model or decentralized model is used.
- Ensure there are no backdoors in the data validation model.
- **Golden Rule: All external input, no matter what it is, is examined and validated.**





Mitigar (Cont.)

Error Handling/Information leakage:

- Ensure that all method/function calls that return a value have proper error handling and return value checking.
- Ensure that exceptions and error conditions are properly handled.
- Ensure that no system errors can be returned to the user.
- Ensure that the application fails in a secure manner.
- Ensure resources are released if an error occurs

Secure Code Environment:

Logging/Auditing:

- Ensure that no sensitive information is logged in
- Ensure the payload being logged is of a defined
- Ensure no sensitive data can be logged; e.g. co
- Examine if the application will audit the actions
- Ensure successful and unsuccessful authentica
- Ensure application errors are logged.
- Examine the application for debug logging with

- Examine the file structure. Are any components that should not be directly accessible available to the user?
- Examine all memory allocations/de-allocations.
- Examine the application for dynamic SQL and determine if it is vulnerable to injection.
- Examine the application for "main()" executable functions and debug harnesses/backdoors.
- Search for commented out code, commented out test code, which may contain sensitive information.
- Ensure all logical decisions have a default clause.
- Ensure no development environment kit is contained on the build directories.
- Search for any calls to the underlying operating system or file open calls and examine the error possibilities.

Cryptography:

- Ensure no sensitive data is transmitted in the cl
- Ensure the application is implementing known g

Session Management:

- Examine how and when a session is created for a user, unauthenticated and authenticated.
- Examine the session ID and verify if it is complex enough to fulfill requirements regarding strength.
- Examine how sessions are stored: e.g. in a database, in memory etc.
- Examine how the application tracks sessions.
- Determine the actions the application takes if an invalid session ID occurs.
- Examine session invalidation.
- Determine how multithreaded/multi-user session management is performed.
- Determine the session HTTP inactivity timeout.
- Determine how the log-out functionality functions.





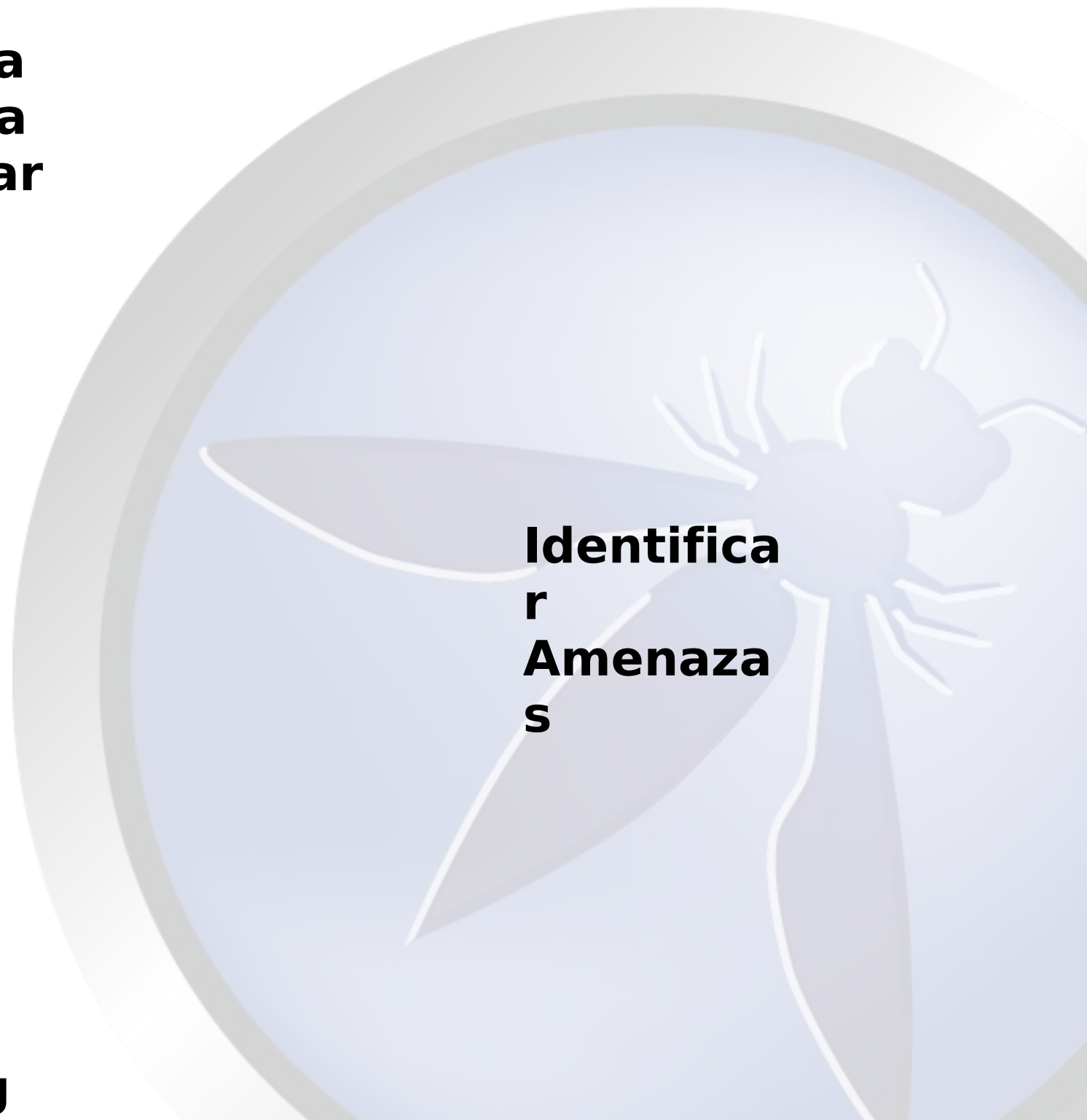
Validar

**Dia
gra
mar**

**Va
lid
ar**

**Identifica
r
Amenaza
s**

**Mi
tig
ar**





Validar (Cont.)





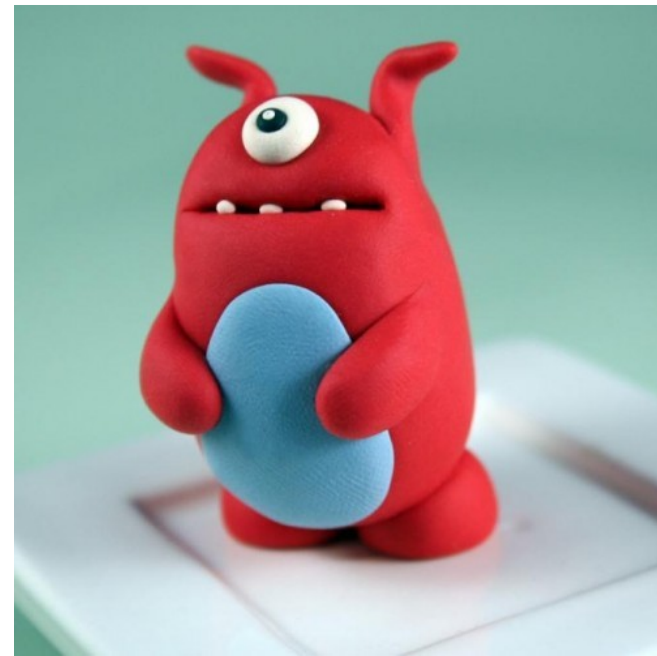
Resumen & Conclusiones





Resumen

- Qué es el Modelado de Amenazas?
- Para qué sirve?
- Por qué llevar adelante este proceso?
- En que momento debo llevar a cabo un MdA?
- Como lo llevo adelante?
 - Diagramar
 - Identificar Amenazas
 - Mitigar
 - Validar





Referencias y Lecturas Complementarias

- Threat Modeling: Uncover Security Design Flaws Using The STRIDE Approach
- <http://msdn.microsoft.com/msdnmag/issues/06/11/ThreatModeling/default.aspx>
- The Security Development Lifecycle: SDL: A Process for Developing Demonstrably More Secure Software (Howard, Lipner, 2006) "Threat Modeling" chapter
- <http://www.microsoft.com/mspress/books/authors/auth8753.aspx>
- Application Threat Modeling (OWASP)
- https://www.owasp.org/index.php/Application_Threat_Modeling
- Threat Risk Modeling (OWASP)
- https://www.owasp.org/index.php/Threat_Risk_Modeling
- Microsoft Security Development Lifecycle Core Training classes
- <http://www.microsoft.com/en-us/download/details.aspx?id=16420>
- Fotos de Plastilina
- <http://www.google.com.ar/search?q=plastilina&hl=es-419&prmd=imvns&source=634>



Preguntas?



Gracias!!

hracciatti@siclabs.com
www.siclabs.com
@my4ng3l