Erstellen sicherer ASP.NET-Anwendungen

Authentifizierung, Autorisierung und sichere Kommunikation

Kapitel 1 - Einführung

J.D. Meier, Alex Mackman, Michael Dunner und Srinath Vasireddy

Microsoft Corporation

Oktober 2002

Auf der Orientierungsseite finden Sie einen Ausgangspunkt und eine vollständige Übersicht zum *Erstellen sicherer ASP.NET-Anwendungen*.

Zusammenfassung

Dieses Kapitel definiert Anwendungsbereich und Aufbau des Handbuchs und beschreibt dessen Ziele. Darüber hinaus werden die grundlegenden Terminologieaspekte vorgestellt und mehrere wichtige Richtlinien dargestellt, die für die in den späteren Kapiteln beschriebenen Informationen gelten.

Inhalt

Das Erstellen von sicheren verteilten Webanwendungen bildet eine Herausforderung. Die Anwendung ist nur so sicher wie das schwächste Glied. Bei verteilten Anwendungen gibt es viele flexible Teile. Das sichere Zusammenwirken dieser Teile erfordert produkt- und technologieübergreifendes Wissen.

Sie müssen viele Dinge beachten – verschiedene Technologien integrieren, das technologische Wissen auf dem aktuellen Stand halten und dem Mitbewerber immer einen Schritt voraus sein. Können Sie sich den Aufwand leisten, wenn Sie noch nicht wissen, wie sichere Anwendungen entwickelt werden? Genauer gefragt, können Sie es sich leisten, es nicht zu tun?

Die vernetzte Landschaft

Können Sie Ihr Wissen über das Erstellen von sicheren Anwendungen heute schon beim Entwickeln von .NET-Webanwendungen nutzen? Können Sie Ihr Wissen in der heutigen Landschaft webbasierter verteilter Anwendungen einsetzen, in der Webdienste Unternehmen mit Unternehmen (B2B) und Unternehmen mit Kunden (B2C) verbinden und Anwendungen unterschiedlich vielen Benutzern zugänglich sind, beispielsweise Benutzern in Intranets, Extranets und im Internet?

Betrachten Sie einige fundamentalen Merkmale der vernetzten Landschaft:

- Webdienste verwenden Standards wie SOAP, XML (Extensible Markup Language) und HTTP (Hypertext Transport Protocol), übertragen aber potenziell vertrauliche Information in der Regel unverschlüsselt.
- B2C-Internetanwendungen übermitteln vertrauliche Daten über das Web.
- B2B-Extranetanwendungen verwischen die Vertrauenslinien und lassen den Aufruf von Anwendungen durch andere Anwendungen in Partnerunternehmen zu.
- Intranetanwendungen sind aufgrund der sensiblen Natur von Gehalts- und Personalanwendungen nicht ohne Risiko. Solche Anwendungen sind insbesondere durch unseriöse Administratoren und verärgerte Mitarbeiter verwundbar.

Anwendungsbereich

In diesem Handbuch werden hauptsächlich die folgenden Themen behandelt:

- Authentifizierung (zum Identifizieren der Clients der Anwendung)
- Autorisierung (zum Bereitstellen der Zugriffssteuerung für diese Clients)
- Sichere Kommunikation (zum Sicherstellen, dass Nachrichten vertraulich bleiben und nicht durch unberechtigte Teilnehmer verändert werden)

Warum Authentifizierung, Autorisierung und sichere Kommunikation?

Sicherheit ist ein weites Thema. Durch Forschungen wurde belegt, dass durch den frühen Entwurf von Authentifizierung und Autorisierung ein hoher Prozentsatz der Verwundbarkeit von Anwendungen beseitigt wird. Die sichere Kommunikation ist ein integraler Bestandteil bei der Sicherung verteilter Anwendungen zum Schutz vertraulicher Daten, wie z. B. Anmeldeinformationen, die von und zu der Anwendung sowie zwischen den Anwendungsebenen übertragen werden.

Ziele des Handbuchs

Dieses Handbuch bildet keine Einführung in das Thema Sicherheit und keine Sicherheitsreferenz für das Microsoft .NET Framework. Dafür steht das .NET Framework Software Development Kit (SDK) in MSDN zur Verfügung. Einzelheiten finden Sie im Abschnitt "Referenz" des Handbuchs. Das Handbuch beginnt dort, wo die Dokumentation endet. Es enthält szenariobasierte Vorgehensweisen, in denen Empfehlungen und bewährte

Techniken beschrieben werden, die vor Ort, durch Kundschaft und von den Produktteams bei Microsoft ermittelt wurden.

Die Informationen in diesem Handbuch sollen die folgenden Verfahren verdeutlichen:

- Erhöhen der Sicherheit für Anwendungen.
- Identifizieren, wann und wo eine Authentifizierung durchgeführt werden soll.
- Identifizieren, wann und wo eine Autorisierung durchgeführt werden soll.
- Identifizieren, wann und wo eine sichere Kommunikation zur der Anwendung (von den Endbenutzern) und zwischen den Anwendungsebenen erforderlich ist.
- Identifizieren häufiger Fallen und deren Vermeidung.
- Identifizieren der wichtigsten Risiken und deren Milderung in Bezug auf Authentifizierung und Autorisierung.
- Vermeiden der Öffnung der Sicherheit aus reinen Gründen der Funktionsfähigkeit.
- Identifizieren, nicht nur *wie*, sondern auch *wann* die verschiedenen Sicherheitsfeatures verwendet werden.
- Beseitigen von Furcht, Unsicherheit und Zweifel.
- Fördern optimaler Vorgehensweisen und vorhersagbarer Ergebnisse.

Zielgruppe des Handbuchs

Dieses Handbuch wurde modular konzipiert. Sie können also die Kapitel auswählen, die Sie lesen möchten. Wenn Sie beispielsweise mehr über die tief greifenden Sicherheitsfeatures lernen möchten, die von einer bestimmten Technologie bereitgestellt werden, können Sie direkt zu Teil III des Handbuchs übergehen (Kapitel 8 bis 12), der detaillierte Informationen zu ASP.NET, Enterprise Services, Web Services, .NET Remoting und Datenzugriff enthält.

Sie können jedoch auch die Anfangskapitel (Kapitel 1 bis 4) in Teil I des Handbuchs zuerst lesen, da diese Ihnen beim Verstehen des Sicherheitsmodells und dem Identifizieren der grundlegenden Technologien und Sicherheitsdienste helfen. Anwendungsarchitekten sollten auf jeden Fall Kapitel 3 lesen, das einige wichtige Einblicke in den Entwurf einer Authentifizierungs- und Autorisierungsstrategie bietet, die alle Ebenen der Webanwendung umfasst. Teil I stellt die grundlegenden Informationen bereit, mit denen Sie aus dem Rest des Handbuchs maximalen Nutzen ziehen können.

Die Kapitel über Intranet, Extranet und Internet (Kapitel 5 bis 7) in Teil II des Handbuchs zeigen, wie bestimmte Anwendungsszenarien gesichert werden. Wenn Sie die Architektur und das Bereitstellungsmuster kennen, das von der Anwendung verwendet wird, können Sie sich in diesem Teil des Handbuchs über die relevanten Sicherheitsaspekte und die erforderlichen grundlegenden Konfigurationsschritte informieren, um spezielle Szenarien zu sichern.

Schließlich helfen Ihnen die zusätzlichen Informationen und Referenzunterlagen in Teil IV des Handbuchs, bestimmte Technologiebereiche besser zu verstehen. Dieser Teil enthält auch eine Artikelbibliothek mit Vorgehensweisen, anhand derer Sie in möglichst kurzer Zeit funktionierende Sicherheitslösungen entwickeln können.

Aufbau des Handbuchs

Dieses Handbuch ist in vier Teile gegliedert. Das Ziel ist eine logische Aufteilung, damit Sie die Inhalte leichter auswählen können.

Teil I, Sicherheitsmodelle

In Teil I des Handbuchs werden die Grundlagen für den Rest des Handbuchs gelegt. Wenn Sie mit den Konzepten, Richtlinien und Technologien vertraut sind, die in Teil I vorgestellt werden, können Sie aus dem Rest des Handbuchs maximalen Nutzen ziehen. Teil I enthält die folgenden Kapitel:

- Kapitel 1, "Einführung"
- Kapitel 2, "Sicherheitsmodell für ASP.NET-Anwendungen"
- Kapitel 3, "Authentifizierung und Autorisierung"
- Kapitel 4, "Sichere Kommunikation"

Teil II, Anwendungsszenarien

Die meisten Anwendungen können als Intranet-, Extranet- oder Internetanwendungen kategorisiert werden. Dieser Teil des Handbuchs stellt einige gebräuchliche Anwendungsszenarien da, die jeweils unter die oben genannten Kategorien fallen. Die charakteristischen Merkmale der einzelnen Szenarien werden beschrieben, und die möglichen Sicherheitsrisiken werden analysiert.

Anschließend wird das Konfigurieren und Implementieren der geeignetesten Strategie für Authentifizierung, Autorisierung und sichere Kommunikation im jeweiligen Anwendungsszenario aufgezeigt. Jedes Szenario enthält außerdem Abschnitte mit einer detaillierten Analyse, häufigen Fallstricken, die zu beachten sind, und häufig gestellten Fragen (Frequently Asked Questions oder FAQ). Teil II enthält die folgenden Kapitel:

- Kapitel 5, "Intranetsicherheit"
- Kapitel 6, "Extranetsicherheit"
- Kapitel 7, "Internetsicherheit"

Teil III, Sichern der Ebenen

Dieser Teil des Handbuchs enthält detaillierte Informationen in Bezug auf die einzelnen Ebenen und Technologien sicherer .NET-vernetzter Webanwendungen. Teil III enthält die folgenden Kapitel:

- Kapitel 8, "ASP.NET-Sicherheit"
- Kapitel 9, "Enterprise Services-Sicherheit"
- Kapitel 10, "Webdienstsicherheit"
- Kapitel 11, ".NET Remoting-Sicherheit"
- Kapitel 12, "Datenzugriffssicherheit"

In jedem Kapitel finden Sie eine kurze Übersicht über die Sicherheitsarchitektur der betreffenden Technologie. Für jede Technologie werden die Authentifizierungs- und Autorisierungsstrategien sowie die konfigurierbaren Sicherheitsoptionen, die programmatischen Sicherheitsoptionen und praktische Empfehlungen für die jeweilige Strategie erläutert.

Jedes Kapitel enthält Richtlinien und Informationen, anhand derer Sie die geeigneteste Option für Authentifizierung, Autorisierung und sichere Kommunikation der jeweiligen Technologie auswählen und implementieren können. Darüber hinaus bietet jedes Kapitel spezielle Zusatzinformationen für die betreffende Technologie. Alle Kapitel enden mit einer kurzen Zusammenfassung der Empfehlungen.

Teil IV, Referenz

Dieser Referenzteil des Handbuchs beinhaltet ergänzende Informationen, die Ihnen beim besseren Verstehen der Techniken, Strategien und Sicherheitslösungen helfen, die in den vorangehenden Kapiteln erläutert wurden. Detaillierte Vorgehensweisen bieten schrittweise Verfahren, anhand derer Sie spezielle Sicherheitslösungen implementieren können. Dieser Teil enthält die folgenden Informationen:

- Kapitel 13, "Problembehandlung bei der Sicherheit"
- "Vorgehensweisen"
- "Grundkonfiguration"
- "Konfigurationsspeicher und -tools"
- "Referenzliste"
- "Vorgehensweisen"
- "Die technischen Grundlagen"
- "ASP.NET-Identitätsmatrix"
- "Kryptographie und Zertifikate"
- "ASP.NET-Sicherheitsmodell"
- "Glossar"

Terminologie

In diesem Abschnitt wird die grundlegende Terminologie für die Sicherheit vorgestellt, die in diesem Handbuch verwendet wird. Obwohl im Abschnitt "Referenz" dieses Handbuchs ein umfassendes Glossar der Terminologie bereitgestellt wird, sollten Sie die folgenden Begriffe unbedingt kennen:

- **Authentifizierung** – Die positive Identifizierung der Clients der Anwendung; Clients können Endbenutzer, Dienste, Prozesse oder Computer umfassen.
- **Autorisierung** – Definieren der sichtbaren Elemente und zulässigen Funktionen in der Anwendung für authentifizierte Clients.
- **Sichere Kommunikation** – Sicherstellen, dass Nachrichten bei der Übertragung in den Netzwerken vertraulich bleiben und nicht geändert werden.
- **Identitätswechsel** – Diese Technik wird von einer Serveranwendung verwendet, um im Namen eines Clients auf Ressourcen zuzugreifen. Für Zugriffsüberprüfungen, die vom Server durchgeführt werden, wird der Sicherheitskontext des Clients verwendet.
- **Delegierung** – Eine erweiterte Form des Identitätswechsels, mit der ein Serverprozess, der eine Aufgabe im Namen eines Clients ausführt, auf die Ressourcen eines Remotecomputers zugreifen kann. Diese Fähigkeit wird durch die Kerberos-Integration in Microsoft® Windows® 2000 und späteren Betriebssystemen bereitgestellt. Bei einem herkömmlichen Identitätswechsel (z. B. durch NTLM) ist lediglich ein einziger Netzwerkhop möglich. Wenn der NTLM-Identitätswechsel verwendet wird, erfolgt dieser einzelne Hop zwischen dem Client und den Servercomputern, sodass der Server bei einem Identitätswechsel auf den Zugriff auf lokale Ressourcen beschränkt ist.
- **Sicherheitskontext** – Sicherheitskontext ist ein generischer Begriff, mit dem die Gesamtheit der Sicherheitseinstellungen bezeichnet wird, die das sicherheitsrelevante Verhalten eines Prozesses oder Threads beeinflussen. Die Attribute der Anmeldesitzung und des Zugriffstokens eines Prozesses bilden zusammen den Sicherheitskontext des Prozesses.
- **Identität** – Identität bezieht sich auf ein Merkmal eines Benutzers oder Dienstes, mit dem eine eindeutige Identifizierung möglich ist. Oftmals handelt es sich dabei um einen Anzeigenamen, in der Regel in der Form Autorität/Benutzername.

Richtlinien

Es gibt eine Reihe von Richtlinien, die für alle in späteren Kapiteln dargestellten Informationen gelten. Diese sind nachstehend zusammengefasst:

- **Machen Sie sich den Grundsatz der möglichst wenigen Rechte zu eigen**. Prozesse, die Skripts oder Code ausführen, sollten mit einem Konto mit möglichst wenigen Rechten ausgeführt werden, um den potenziellen Schaden bei einer Gefährdung des Prozesses zu begrenzen. Wenn ein böswilliger Benutzer Code in einen Serverprozess einbringen kann, bestimmen die Rechte, die dem betreffenden Prozess gewährt wurden, im Wesentlichen die Art des Vorgangs, den der Benutzer ausführen kann. Code, der ein zusätzliches Vertrauen (und höhere Rechte) erfordert, sollte in getrennten Prozessen isoliert werden.

  Das ASP.NET-Team hat bewusst entschieden, das ASP.NET-Konto mit möglichst geringen Rechten (Konto ASPNET) auszuführen. In der Betaversion des .NET Framework wurde ASP.NET als SYSTEM ausgeführt, eine deutlich geringere Sicherheitseinstellung.
- **Verwenden Sie eine tief greifende Verteidigung**. Fügen Sie in allen Schichten und Subsystemen ihrer Anwendung Prüfpunkte ein. Die Prüfpunkte bilden die Gatekeeper. Diese stellen sicher, dass nur authentifizierte und autorisierte Benutzer auf die nächste nachgeschaltete Schicht zugreifen können.
- **Vertrauen Sie nicht der Benutzereingabe**. Die Anwendungen sollten alle Benutzereingaben gründlich überprüfen, bevor Operationen mit dieser Eingabe durchgeführt werden. Die Überprüfung kann das Herausfiltern von Sonderzeichen enthalten. Diese Vorsichtsmaßnahme schützt die Anwendung vor versehentlichem Missbrauch oder vorsätzlichen Angriffen durch Personen, die versuchen, böswillige Befehle in das System einzubringen. Bekannte Beispiele umfassen SQL Injection-Angriffe, Skript Injection und Pufferüberlauf.
- **Verwenden Sie sichere Standardeinstellungen**. Bei Entwicklern ist es gängige Praxis, reduzierte Sicherheitseinstellungen zu verwenden, damit eine Anwendung überhaupt funktioniert. Wenn die Anwendung Features erfordert, die Sie zwingen, die Standardsicherheitseinstellungen zu reduzieren oder zu ändern, sollten Sie die Auswirkungen überprüfen und verstehen, bevor Sie die Änderung durchführen.
- **Vertrauen Sie nicht auf Sicherheit durch Verschleierung**. Wenn Sie versuchen, geheime Informationen durch Verwenden von irreführenden Variablennamen oder Speichern in unkonventionellen Dateiverzeichnissen zu verbergen, wird keine Sicherheit erreicht. Sie sollten stattdessen Features der Plattform oder bewährte Techniken verwenden, um die Daten zu sichern.
- **Führen Sie Überprüfungen am Gate durch**. Sie müssen den Sicherheitskontext eines Benutzers bei der Überprüfung der Autorisierung nicht immer bis zum Back-End fließen lassen. In einem verteilten System ist dies oftmals nicht die optimale Möglichkeit. Die Überprüfung des Clients am Gate bezieht sich auf die Autorisierung des Benutzers am ersten Authentifizierungspunkt (beispielsweise in der Webanwendung auf dem Webserver) und die Ermittlung der Ressourcen und Operationen (die möglicherweise von nachgeschalteten Diensten bereitgestellt werden), auf die der Benutzer zugreifen darf.

  Wenn Sie stabile Authentifizierungs- und Autorisierungsstrategien am Gate entwerfen, können Sie die Notwendigkeit umgehen, den Sicherheitskontext des ursprünglichen Aufrufers durch die gesamte Datenebene der Anwendung delegieren zu müssen.
- **Gehen Sie davon aus, dass externe Systeme nicht sicher sind**. Gehen Sie nicht davon aus, dass Ihnen Verantwortung für die Sicherheit abgenommen wird, wenn ein System nicht in Ihrem eigenen Verantwortungsbereich liegt.
- **Reduzieren Sie die Oberfläche**. Vermeiden Sie das Offenlegen von nicht benötigten Informationen. Dadurch werden eventuell Türen geöffnet, die zu weiteren Sicherheitsrisiken führen. Weiterhin sollten Sie Fehler ordnungsgemäß behandeln. Legen Sie nicht mehr Informationen als erforderlich offen, wenn eine Fehlermeldung an den Benutzer zurückgegeben wird.

- **Wechseln Sie bei Fehlern in einen sicheren Modus**. Stellen Sie bei einem Fehler der Anwendung sicher, dass vertrauliche Daten nicht ungeschützt bleiben. Stellen Sie in Fehlermeldungen außerdem nicht zu viele Details bereit, d. h., lassen Sie Details weg, die einem Angreifer helfen können, eine verwundbare Stelle in der Anwendung zu finden. Schreiben Sie detaillierte Fehlerinformationen in das Windows-Ereignisprotokoll.
- **Denken Sie daran, dass die Anwendung nur so sicher ist wie ihr schwächstes Glied**. Sicherheit betrifft alle Anwendungsebenen.
- **Deaktivieren Sie, was nicht verwendet wird**. Sie können potenzielle Angriffspunkte entfernen, indem Sie von der Anwendung nicht benötigte Module und Komponenten deaktivieren. Wenn die Anwendung beispielsweise keinen Ausgabecache verwendet, sollten Sie das Ausgabecachemodul von ASP.NET deaktivieren. Wenn später ein Sicherheitsrisiko im Modul festgestellt wird, besteht keine Gefahr für die Anwendung.

Zusammenfassung

In diesem Kapitel wurden einige grundlegende Informationen bereitgestellt, die als Vorbereitung für den Rest des Handbuchs dienen. Es wurden die Ziele des Handbuchs beschrieben und die Gesamtstruktur dargestellt. Sie müssen die Schlüsselterminologie und Grundsätze unbedingt kennen, die in diesem Kapitel vorgestellt wurden, da sie in den folgenden Kapiteln umfassend verwendet werden und darauf verwiesen wird.