IceWall MCRP

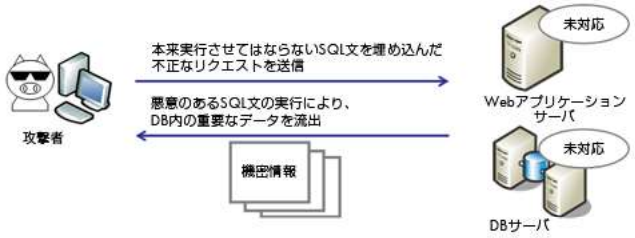
IceWall技術レポート：IceWall MCRP 2.1 SP1の新機能　～SQLインジェクション対応機能～

Webシングルサインオン機能を核としたセキュリティーソリューションを提供する IceWallファミリーの中でも、高速でセキュアなリバースプロキシ機能を提供するIceWall MCRPに注目が集まっています。
本技術レポートでは、IceWall MCRP 2.1 SP1より追加されたSQLインジェクション対応機能の特長および具体的な設定例を紹介していきます。

概要

IceWall MCRP 2.1 SP1では新しいセキュリティ機能としてSQLインジェクション攻撃から防御できるようになりました。SQLインジェクション対応機能を使用することで、外部の悪意を持った攻撃者（ハッカー）が、リレーショナルデータベースを不正に操作することを目的にWebアプリケーションに送ってくる可能性のある有害な文字列パターンを検出し、情報の漏えいや改ざんを未然に防止することが可能です。

●IceWall MCRP導入前

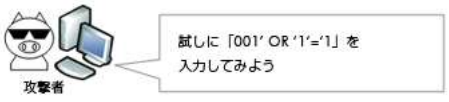


●IceWall MCRP導入後

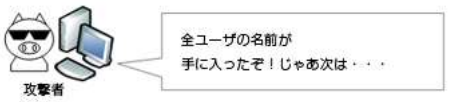
SQLインジェクション攻撃とは

IPA（情報処理推進機構）の発行する「ソフトウエア等の脆弱性関連情報に関する届出状況」によれば、近年においてもSQLインジェクションはいまだに発見される脆弱性の中でも上位を占めており、対策がされないまま脆弱性が放置されたままになっているサイトが非常に多くなっているということです。SQLインジェクションの特徴として、万が一その被害を受けた場合に、機密情報の漏洩など企業や団体にとって致命的な損害を招く可能性が高い事があげられます。

■Webアプリケーションで用意されているSQL文
SELECT userName from auth_table where userID='ユーザ入力文字列';
※userIDがユーザ入力の文字列とマッチした場合、該当行のuserNameを返すSQL文



■Webアプリケーションで実行されるSQL文
SELECT userName from auth_table where userID='001' OR '1'='1';
※全ユーザに対して'1'='1'が真になってしまい、全ユーザのuserNameが出力



システム用のテーブルからテーブル名を検索したり、返って来たエラーをもとにテーブル構造を推測するなど、実際の攻撃手順はもっと複雑のようです。

SQLインジェクションの具体的な手法にはいくつかありますが、基本的に何らかの悪意を持ったユーザがWebアプリケーションへSQL文を含んだ文字列を送信し、Webアプリケーションを通じてデータベース上でそのSQL文を実行させて、本来公開できない情報を入手したり、データベースの情報を改ざんしたりする点で共通しています。

SQLインジェクションへの対応として、本来はデータベース側で実行されるSQLをあらかじめ定義することや、アプリケーション内で入力文字列やHTTPヘッダ（Cookie等）内の特殊文字を適切にエスケープしておくといった対策を取るのが理想です。しかし、既存のアプリケーションの改修は簡単ではありませんし、防御ロジックを追加することで別のバグや性能劣化を招く恐れがあります。また、そもそもソースコードが残っていない場合や、パッケージ製品等のように開発元からのパッチの適応やバージョンアップをしないと問題が解消されない、あるいはサポートが切れていてそれらの対応もできないというようなケースも少なくありません。

IceWall MCRPのSQLインジェクション対応機能

このような場合に、IceWall MCRPをWebアプリケーションの前段に配置して、不正な文字列をフィルタリングするよう設定すると、アプリケーションの改修をすることなく、またパフォーマンスの劣化も引き起こすことなく、SQLインジェクション攻撃への防御を行う事ができます。

IceWall MCRPのSQLインジェクション対応機能では、SQLインジェクション対応機能用設定ファイル（injection.conf）に以下の各項目を定義します。

- 検査対象とするURL
- 送信データの種類（QUERY、POSTデータ、HTTPヘッダ、URL）
- 検査対象とする属性名
- フィルタリングする文字列（正規表現による記述が可能

これらを以下のように記述をします。
＜対象URL＞,＜送信データ種類＞,＜属性名＞＝＜フィルタリング文字列＞

設定例①
「/service1」以下のディレクトリ（MCRPがWebアプリケーションサーバのURLに変換）に送信されるQuery Stringのうち、useridという属性に含まれる、シングルクォーテーションをフィルタリングしたい場合は、以下の設定をします。

/service/,QUERY,userid=|%27

ここで、「|」は正規表現における「または」を意味し、%27はシングルクォーテーションがURLエンコードされた値となります。
この設定によって、悪意のあるユーザがシングルクォーテーションを含んだ文字列をuseridという属性に埋め込んで送信してきた場合、MCRPがエラー画面を表示します。

設定例②
先ほどの設定例①ではシングルクォーテーションをフィルタリングしましたが、アプリケーションによっては、シングルクォーテーションが入力される場合（外国人の名前等）が想定される場合があります。そうした場合は、例えばSQLで使用する予約語などをあらかじめ定義しておくことでSQLインジェクションの実行を防ぐことができます。

/SQL/,POST,=or|select|insert|update|OR|SELECT|INSERT|UPDATE

この設定では、「/SQL」以下のディレクトリに、POSTデータ内の任意の属性（設定を省略した場合はすべての属性が検査対象となります）内に、SQLで使用されるOR、SELECT、UPDATE、INSERTが含まれている場合にMCRPがエラー画面を表示します。
当然、これらの文字列がアプリケーション上で使用される可能性もありますので、その場合は設定例①と②を、正規表現を使ってうまく組み合わせる等の対応が必要になる場合もあるでしょう。

具体的な設定例③
ごく最近、新しい攻撃パターンとして、フォームの入力文字列ではなく、ユーザに発行されるCookieの情報の中にSQL文を埋め込み、データベースの改ざんなどを行う手法が出てきています。ユーザに発行したcookieを都度データベース内に保存・更新してセッション管理等を行うアプリケーションを狙った攻撃で、実際に商用サイトで被害も出ているようです。IceWall MCRPではcookieをはじめとするHTTPヘッダ内の情報に含まれる文字列もフィルタリングができます。例えば、cookie内にシングルクォーテーションが含まれていた場合、フィルタリングを行う設定は以下となります。

/web01/,HEADER,Cookie=|%27

おわりに

このように、IceWall MCRPのSQLインジェクション対応機能により、様々なパターンのSQLインジェクション攻撃を防御することができます。システムのトータルセキュリティの向上にIceWall MCRPをぜひ一度ご検討ください。
また、SQLインジェクションに限らず、Webを通じた攻撃手法は日々進化してきており、定期的にシステムのセキュリティのチェックや対策の見直しを行うことが重要です。そうした定期的なセキュリティ検査には、HPの提供する統合アプリケーションセキュリティ検査ソリューション「Application Security Center」の脆弱性検査製品「HP WebInspect software」がおすすめです。

※ HP WebInspect softwareは、Dailyで更新される脆弱性情報データベースに基づいて自動化された検査を実施するため、常に最新で高精度の検査を行うことができ、これまで複雑で高いスキルが必要とされていたセキュリティ検査を容易に実現できます。

2008.10.24 日本ヒューレット・パッカード コンサルティング・インテグレーション統括本部　テクニカルコンサルタント　徳永　拓