

集された影響算出部において未対応のパターンのソフトウェア情報を修正する機能である。当修正機能を使用した未対応のパターンに対する修正の流れを述べる(図3)。PC1に導入したエージェントからソフトウェア情報がIT資産管理部に送信される。送信されたソフトウェア情報は修正機能で修正履歴と比較される。修正履歴に無い場合、未対応のパターンに対し、当修正機能を用いてWebUIからシステム管理者が修正作業を行う修正した内容は修正履歴に記録される。次に、PC2に導入したエージェントからPC1と同様のソフトウェア情報がIT資産管理部に送信される。PC2から送信されたソフトウェア情報は、修正機能が修正履歴を参照し、同様のパターンがあれば修正履歴に従って修正を行う。

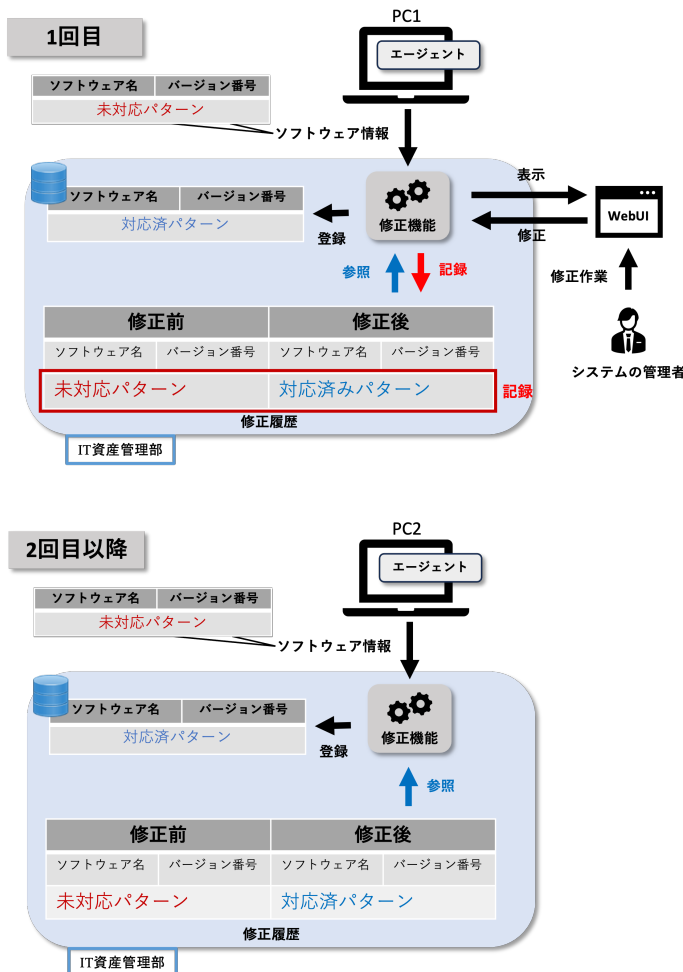


図 2: ソフトウェア情報の自動修正機能

4 評価

開発したソフトウェア情報修正機能を使用した時のBEYONDの性能評価について述べる。評価対象として評価結果を図3に示す。拡張したエージェントの性能評価について述べる。エージェント実行時のCPU使用率は0.2

	再現率	適合率	F値
LDAPサーバ2修正前	0.577	0.94	0.715
LDAPサーバ2修正後	0.994	0.954	0.973
SMTPサーバ3	0.994	0.954	0.973

図 3: 評価結果

5 終わりに

本稿では、BEYOND運用のために各機構に対してWebAPIを実装し、各DBの情報を他の機構でも扱えるようにすることでBEYONDを1つのシステムとして統合した。また、ユーザがBEYONDを利用するためのWebUIを実装した。これにより、BEYONDを一つのシステムとして扱うことが出来るようになった。さらに、ソフトウェア修正機能を実装することで、脆弱性情報の再現率を向上させることができた。

参考文献

- [1] 中村友昭 他, “脆弱性情報を用いたセキュリティ保護システム“BEYOND”の開発” 大学ICT推進協議会 2022年度年次大会論文集, 13PM2B-3, 2022