情報処理学会論文誌 Vol. 51 No. 2 529–541 (Feb. 2010)

ユーザビリティ工学に基づくユーザビリティとセキュリティを両立させたセキュリティスキャナのインタフェースの開発と評価

吉本道隆[†1,*1] 加藤貴司[†1]
ベッド バハドゥール ビスタ[†1] 高田豊雄[†1]

現在流通している個人向けセキュリティ製品は，開発者が想像しているユーザ像と実際のユーザとの間に差異があるため，すべてのユーザにとって必ずしも使いやすいとはいえない．実際，多くのユーザはセキュリティに関する知識がなく，セキュリティ製品の導入を躊躇する場合がある．加えて，従来より，多くのユーザや開発者はセキュリティ製品のユーザビリティとセキュリティ（機能性や信頼性など）はトレードオフの関係にあり，共存することができないと信じ込んでいるとされている．本論文ではユーザビリティとセキュリティの両立したセキュリティ製品の開発を目指し，ユーザビリティ工学の諸手法を既存のセキュリティ製品のインタフェースに対して適切に適用することにより両立を達成する一手法について述べる．本論文では作業例題としてセキュリティスキャナについて取り上げる．開発と評価に際してユーザ中心設計と ISO 9241-11 で示されるユーザビリティ3 要素に着目する．まず，ユーザの実際の要求を把握するためにコンテキストインタビューを行う．次に，現在配布されているセキュリティスキャナに潜在するユーザビリティの問題点について調査する．開発の途中段階においてフィードバックを得るために被験者を用いたユーザビリティ評価を 5 回行った．最終的に総括的評価を行い，客観的な視点から高いユーザビリティを保てていることを示す．さらに意識調査を行い，このようなシステムが提供されることによって，ユーザのセキュリティに対する意識も変わることを示す．

Development and Evaluation of Interface of Security Scanner with Usability and Security Based on Usability Engineering

Michitaka Yoshimoto,[†1,*1] Takashi Katoh,[†1]
Bhed Bahadur Bista[†1] and Toyoo Takata[†1]

Presently available security products are not usable for all users because there is a gap between the image of users the product developers have and the users who actually use the products. In reality many users do not know anything about security and hesitate to implement the security products. Additionally, it has been believed that usability and security (functionality, reliability and so on) of the products do not coexist. For examining the coexistence of usability and security, in this paper, we propose a method for developing interfaces of security products with usability and security by applying a proper combination of several methodologies in usability engineering to the existing security product. We consider security scanner and develop the system as a working example. The system we develop is in accordance both with user centered design and three elements of usability shown in ISO 9241-11. First, we conduct context interview for grasping user's requirements. Next, we grasp underlying problems of presently available security scanners. We perform usability evaluation five times with subjects in intermediate step of development for receiving feedback. Finally we perform summative evaluation and we show that high usability has been realized from an objective viewpoint. Moreover, we conduct attitude survey, and show that user's awareness about security can be increased if they are provided with products which have high usability.

1. 背　　景

今日，常時接続を用いたインターネットユーザ数は爆発的な増加傾向にあり，接続端末も多種多様化している．同時に不正アクセス内容も巧妙かつ悪質となり，件数も増加傾向にある．IPA による報告によると[1)]，不正アクセスのターゲットは個人ユーザから大企業まで，相手を問わない傾向になっており，すべてのユーザに適切なセキュリティに関する対策が求められている．もしユーザが基本的なセキュリティ対策を行っていれば，ほとんどの不正アクセスを未然に防ぐことができたと報告されているが，ユーザはセキュリティに対する知識を持ち合わせているとは必ずしも限らない．別の報告によると[2)]，40%のユーザは不正攻撃によって起こりうる被害について想像することができないと回答したと報告されている．特に，この回答を行ったユーザはセキュリティ製品の必要性に関する質問において多くのユーザが “どれほどのセキュリティ対策を行わなければならないか分からない” と回答している．さらにその報告では，セキュリティ対策を “行っている” という回答率がより低く，“実施しているかどうか分からない” という回答率がより高いと報告されている．加えて，約 70%の

†1 岩手県立大学大学院ソフトウェア情報学研究科
Graduate School of Software and Information Science, Iwate Prefectural University
∗1 現在，清泉女学院大学
Presently with Seisen Jogakuin College

ユーザは “セキュリティホール” という単語の意味を正しく理解できていないと報告されており[3]，たとえセキュリティ技術や情報を提供したとしても，彼らはそういった単語を理解できないという現状である．それにもかかわらず，セキュリティツールにおいてユーザの理解できない難解な専門用語を多用することにより，結果としてユーザにとってユーザブルでないものとなっている．

また専門用語の難解さ以外の問題も存在する．文献 4) によれば，Furnell らは 340 人の被験者を用いて Windows XP と 3 つのアプリケーション（Microsoft Internet Explorer, Word，Outlook Express）のセキュリティに関する操作と設定，たとえばドキュメントやメールの暗号化などについてユーザビリティテストを行ったところ，ほぼすべての被験者が何かしらの困難を感じたと指摘している．

これらの結果，多くのセキュリティ技術やツールが開発・提供されており（未知の攻撃手法やマルウェアなどを除けば）それらの適切な利用によってセキュリティ確保が可能な状態にあるにもかかわらず，現実にはそれらの積極的な導入に至らず，結果として前述の文献 1) にあるとおり既知の脆弱性を悪用した攻撃が後をたたない．すなわち，既存ツールのユーザビリティの低さがセキュリティ対策の遅れを招く結果をもたらしていると考えられる．よって信頼性や機能が十分確保された既存ツールのユーザビリティの改善されたものが提供されればいっそうのセキュリティの確保・向上が期待される．

ほぼすべての不正アクセスはアプリケーションや OS に対して提供されているパッチを適用することや，つねに最新のアップデートを行うことによって未然に防ぐことができる．しかしながら，パッチを適用するだけでは不十分な場合や，ユーザが自身の管理するシステムを完全に把握しているとは限らない場合があり，ユーザは自分の管理する端末に脆弱性が存在するかどうかを自動的に把握できることが望ましい．そこで本論文では “セキュリティスキャナ” について着目する．セキュリティスキャナはユーザの端末に存在する脆弱性を，様々な方法で自動的に調査することを目的としたツールである．

また，専門家らによって，セキュリティの分野においてユーザビリティとセキュリティ（機能性や信頼性など）はトレードオフの関係にあると信じられている[5]．そのため，ユーザビリティ工学に基づいたユーザビリティとセキュリティを両立させたセキュリティツールの開発は進んでいなかった．本論文の目的は，セキュリティ製品のユーザビリティを改善し，そのユーザビリティを評価することによって，ユーザビリティとセキュリティの両立可能性を検証すること，また，ユーザビリティとセキュリティの両方を満足させるための一手法を示すことである．

本論文では，Norman らによって提唱された “ユーザ中心設計” [6] に着目する．通常の開発方法ではセキュリティツール開発者が想像するユーザ像と，実際のユーザの間には差異がある場合があり，開発者が想像するとおりにユーザはそのツールを使いこなすことができない場合が多分にある．また，ユーザビリティ工学において，ユーザの意見の把握を開発の前後に行う従来の方法では潜在する重大なユーザビリティ問題を完全に見つけることができないとされている．ユーザ中心設計において，開発者はシステム開発以前において実際のユーザの要求を把握し，開発の途中段階において開発指標としてユーザの多くの意見を採用する．この方法を用いることによって，開発者は潜在的で重大な問題点をすべて発見することができる．そこでユーザ中心設計をとり入れ，設計段階からユーザの意見を多くとり入れながら開発を行っていく方法をとることによって，実際のユーザの利用状況にあったツール開発が行えるようになる．本研究で提案する手法を適用したセキュリティスキャナではこの手法を採用するとともに，開発したシステムをユーザが効率的かつ効果的に使用可能とするためのユーザサポートデータベースが必要であると判断し，それを新たに構築した．ユーザサポートデータベースにおいては操作や判断を手助けするバルーンヘルプや，発見された脆弱性を効率良く修正するための手助けとなる追記情報などを提供する仕組みをとった．

また，ISO 9241-11 [7] ではユーザビリティ 3 要素が示されている．そこでは，1) 効果的，2) 効率的に，3) 満足に使えることがユーザビリティとして求められており，セキュリティツールを使用するうえにおいても，その 3 要素は満たされているべきであると考えられる．そこで，本論文では総括的評価において被験者を用いたパフォーマンス測定を行い，3 要素について数値化された結果を得ることによって，客観的な評価結果が得られる仕組みをとった．

本論文ではセキュリティに関する知識は必ずしも十分ではないが，コンピュータ操作にはある程度習熟したユーザを対象とする．セキュリティに関する知識を持ち合わせ，コンピュータを自在に使いこなすユーザはどのようなインタフェース提供（たとえそれが CUI でも）がなされていてもそのシステムを使いこなすことができると予想される．それらのユーザをターゲットとしてユーザビリティ改善を行った場合，たとえ改善に失敗していたとしても，改善の度合いが明確には現れにくいためである．また，6 章でも述べるとおり，30 名の被験者を用いたユーザビリティ評価の結果，セキュリティスキャナのようなツールは多くのユーザにとって有用であるが一般的ではなく，セキュリティの知識を持ち合わせないユーザにとってユーザブルではないという結果が出ている．さらに，ユーザビリティ工学において，まったくの初心者は評価対象を評価する以前の問題点につまずく可能性があり

（本研究ではコンピュータの基本操作を行うことが十分にできないなど），ユーザビリティ評価実験の被験者に適さないという見解がある．したがって，そのようなユーザを対象とし，信頼性や機能性などを低下させることなく，彼らにとってユーザブルでないものをユーザブルにする試みを行う．そしてこういったシステムの普及がユーザのセキュリティに対する意識向上につながることを示す．

本論文の構成は以下のとおりである．2 章において，セキュリティスキャナについておよび関連研究について述べ，3 章において，評価方法について述べる．4 章において，既存のセキュリティスキャナの予備調査結果を示し，5 章において，形成的評価の結果を述べる．6 章において，既存のセキュリティスキャナと本論文で提案する手法を適用して得られたセキュリティスキャナの総括的評価を行いその比較結果を示し，セキュリティに関する意識調査結果についても述べる．

2. セキュリティスキャナ

本論文ではユーザビリティの改善を考察する作業例題としてセキュリティスキャナを取り上げる．本章ではセキュリティスキャナについて説明し，セキュリティスキャナの開発やメンテナンスコストについて述べ，関連研究として既存のセキュリティスキャナについて述べる．

2.1 セキュリティスキャナについて

セキュリティスキャナとは，ネットワーク上の端末に対して，その端末の脆弱性の有無を調べるツールである．通常操作しているだけでは分からない脆弱性を発見し，ユーザにその発見された脆弱性とその対処法を通知する．ユーザはセキュリティスキャナの通知内容に従って脆弱性の対処を行う．セキュリティスキャナは既知の脆弱性のほとんどをスキャンすることができる機能性を持ち合わせ，脆弱性の有無の情報を視覚的にユーザに与える．

2.2 ネットワーク型セキュリティスキャナとクライアント型セキュリティスキャナ

本論文では “ネットワーク型セキュリティスキャナ” を取り上げる．ネットワーク型セキュリティスキャナはネットワークを介したクライアントに対して脆弱性診断を行うため，ネットワークに接続されていないアプリケーションや機能については脆弱性を発見することができない．そのようなアプリケーションや機能は “クライアント型セキュリティスキャナ” によってスキャンされるべきであり，本論文ではそれについては取り上げない．なぜならばクライアント型セキュリティスキャナの対象となる脆弱性の多くはそれぞれの国ごとで管理・整備されることが多く，一般性をもった議論が困難である．たとえば JVN#66077895[8] はアンチウィルスソフトの DoS の脆弱性に関する脆弱性レポートであり，日本国内で大きく取り上げられた脆弱性であるが，海外では CVE が CVE-2008-4429[9] として取り上げられているものの概要を述べているのみにすぎず，その他海外の脆弱性情報ベンダにおいてはほとんど議論されていない．

2.3 セキュリティスキャナ開発および維持コスト

セキュリティスキャナを開発・管理するにあたり，非常に多くのコストが必要となる．脆弱性は日々多く発見されており，それに対応するスキャンスクリプトの生成を行わなければならない．また，そのスキャンスクリプトが正常に動作するか，脆弱性の持つ危険性などの情報が変化していないかなどのチェック・反映をつねに行わなければならないため，莫大なコストを必要とする．そこで，セキュリティスキャナは，すでに開発され現在も維持されているセキュリティスキャナを核とし，それに対して処理の命令や結果の受け取りなどの通信を行うシステム開発が望ましいと考えられる．そこで本研究では配布されているフリーウェアのセキュリティスキャナの中で最も普及しており，信頼性が高いとされている[10]，Nessus[11] を核としたセキュリティスキャナを開発することとした．Nessus はたとえば 2009 年 2 月 1 日から 10 日の 10 日間に 524 個ものプラグイン（スキャンスクリプト）の追加・更新を行っている．

2.4 セキュリティスキャナにおける関連研究

本節ではこれまでに提案されているセキュリティスキャナについて概説する．

2.4.1 バージョン情報のみを調べるセキュリティスキャナ

バージョン情報のみを調べるセキュリティスキャナは脆弱なバージョンのアプリケーションを使用していないかを調べ，バージョンアップやパッチ適用などの勧告を行うセキュリティスキャナである．

しかし，いくつかの種類の脆弱性はアプリケーションの設定に起因して起こりうる．たとえば，CVE-2007-3215[12] の脆弱性は PHP のクラスである PHPMailer が sendmail を使用するように設定されていた場合に，悪意のある任意のコードの実行の許可を行ってしまう脆弱性である．バージョン情報のみで脆弱性の有無を判別するセキュリティスキャナの場合，該当の製品のバージョン情報を調べるだけであり，構成や設定まで調べないため，この脆弱性は発見することができないか，該当製品バージョンが更新されるまで誤検知を行う可能性がある．さらに Apache や BIND，sendmail において，サーバ管理者はそれらのバージョン情報をセキュリティ対策として隠蔽・改変を行うことが容易に行える．

日本において，このようなセキュリティスキャナの研究が数々行われている．たとえば

毛利らの提案するセキュリティスキャナ[13] は SMTP（sendmail）や HTTP（Apache），DNS（BIND）サーバの脆弱性に注目する．それぞれのデーモンが出力するバージョン情報などを基に脆弱性診断を行い，結果内容は JVN [14] の提供する脆弱性情報とともに出力される．しかし前述のとおりバージョン情報が隠蔽・改変されたサーバの場合，提案するシステムは正常に診断を行うことができない．

また，藤平らの提案するセキュリティスキャナ[15] は，実際に稼働しているシステムのバージョン情報やネットワーク構成をスキャンしバーチャル空間内に再現し，仮想的な環境下で脆弱性診断を行うシステムである．しかし同様にバージョン情報を正常に読み取ることができなければ正常に動作することができず，前述の CVE-2007-3215 のような脆弱性の場合，仮想空間に再現し診断を行うことは困難である．

2.4.2　構成を調べるセキュリティスキャナ

構成を調べるセキュリティスキャナはネットワークに接続されたすべてのクライアント端末に対して診断を行うことができる．

小手川ら[16] はモバイルエージェントベースのセキュリティスキャナを提案している．このセキュリティスキャナでは MITRE 社が提案する OVAL [17] に着目している．OVAL とは脆弱性情報記述言語であり，SQL や XML 形式で記述されており，OVAL Definitions として公開されている．それをクライアント端末に事前に導入されている OVAL Definition Interpreter へ入力することにより，そのクライアント端末に導入されている OS やアプリケーションの種類や設定などを読み取り，脆弱性診断を行うことができる．そしてその情報をモバイルエージェントが収集し，ネットワークを一元管理するシステムである．しかし，提案されているモバイルエージェントはルータなどのハードウェア機器に対して情報収集を行うことができないため，ネットワーク全体のセキュリティを検査する際には適用できない．

2.4.3　Web ベースのスキャナ

The Inprotect Team [18] は Nessus を核とした Web ベースのセキュリティスキャナである Inprotect を開発し，配布している．Inprotect はウェブブラウザから Nessus が稼働する Web サーバに向けて診断リクエストを発することにより，Nessus を動作させることができるシステムである．しかし，Inprotect から診断要求を行った後，進捗状況などがまったく把握できないため，ユーザは診断結果がいつ得られるのか予測することができない．文献 19) において，プログレスバー表示が人間の生理的指標にどのような影響を与えるかについて述べられており，スキャンの長い間，ユーザに対してシステムが何を行っているか，また作業がいつ終わるかが示されないことはユーザビリティがまったく不十分であるとされている．したがって，ユーザは Inprotect を効果的かつ効率的に，満足に使うことができない可能性が多分にある．

2.5　比較対象とするセキュリティスキャナ

前述のとおり，Nessus はセキュリティスキャナとしては高い信頼性と機能性を持ち合わせている．この研究を始める以前に筆者は Nessus を含めた様々なセキュリティスキャナの調査を行っており，ユーザビリティという観点に着目したとき，それらのセキュリティスキャナに共通したユーザビリティ問題を発見している[20]．たとえばほとんどすべてのセキュリティスキャナは結果表示方法がほとんど同じ（危険度，脆弱性の説明，リスク，解決策，脆弱性のスコアなど）であり，それらによって表示される情報は膨大で平板であり，その情報はセキュリティに対する知識を持ち合わせないユーザを困惑させるだけであることを示した[21]．したがって，あらゆるセキュリティスキャナのユーザビリティ問題を調査し評価の対象とすることは合理的でないと考えられる．そのため，本研究では既存のセキュリティスキャナの評価対象の代表として Nessus の Windows クライアントである NessusWX を取り上げることとした．本論文では Nessus を核としたセキュリティスキャナ開発を行うが，Nessus は前述したとおりユーザビリティ上の問題をかかえている．そこで Nessus のセキュリティ診断上の機能や診断性能は余すことなく維持したまま，ユーザビリティを向上させるシステム開発を行うこととする．そのため，前述のとおり Nessus は核とし手を加えず，そのインタフェースを，情報の提示の方法など，様々な既成概念にとらわれることなく一から設計することとする．まず，本論文では Nessus の Windows クライアントである NessusWX を提案するシステムのプロトタイプであると見立て，ユーザビリティ上の問題点を発見する．そして発見された問題点に対し，繰返し改善法の検討と改善と問題点の再発見をすることによって，ユーザビリティの向上を図る．ただし，単にユーザビリティの向上を図るだけでなく，診断上の機能や性能を完全に維持したまま，ユーザビリティを向上させる手法が求められる．その手法について次章以降において述べる．

3. 評 価 方 法

本章では開発するシステムのユーザビリティの評価方法について述べる．評価は “形成的評価” と “総括的評価” からなり，評価人数は総括的評価においては 1 ターゲットあたり 20 から 30 人程度，形成的評価においては 5 人程度必要であるとされており，Nielsen は 5 人で評価を行うことの妥当性について述べている[22]．

3.1　ユーザ中心設計

前述のとおり，開発者が想像しているユーザ像と実際のユーザとの間に差異があるため，既存のセキュリティスキャナはほとんどのユーザにとってユーザブルでない．そこで本研究のセキュリティスキャナシステム開発では “ユーザ中心設計” に注目した．ユーザ中心設計において，ソフトウェアはユーザの目線でデザインされ，開発プロセスのフレームワークは下記のようになる．

ステップ 1：ユーザの “利用状況” を把握する．
ステップ 2：利用状況から “ユーザニーズ” を探索する．
ステップ 3：ユーザニーズを満たすような “解決案” を作る．
ステップ 4：解決案を “評価” する．
ステップ 5：評価結果をフィードバックし，解決案を “改善” する．
ステップ 6：評価と改善を “繰り返す”．

以上のステップを経ることにより，実際のユーザの使用状況にあった設計・開発が行うことができ，評価を繰り返し行うことによって小さな問題点から重大な問題点までほとんどすべてを発見・改善を行うことができる．

3.2　ISO 9241-11:1998

ユーザビリティに関する規格化された定義には IEC 300，ISO/IEC 9126，FAA 1998 などがあり，文献 23) においていくつか紹介されている．本論文の評価実験では，日本語化され JIS 規格（JIS Z8521）としても採用されている ISO 9241-11:1998 で解説されているユーザビリティ3 要素に着目する．もし，その 3 要素のどれかに問題があれば，そのシステムは “使いものにならない” とされている．以下にそれを示す．

Effectiveness（効果） Accuracy and completeness with which users achieve specified goals.

Efficiency（効率） Resources expended in relation to the accuracy and completeness with which users achieve goals.

Satisfaction（満足度） Freedom from discomfort, and positive attitudes towards the use of the product.

セキュリティツールにおいて，効果的にツールを使うことができなければ脆弱性を完全に取り除くことができない可能性があり，ユーザの端末が致命的な状態のまま放置される可能性が考えられる．また効率良く使えなければ対処に時間がかかるのはもちろん，セキュリティに関する知識のないユーザは途中で使用を中止することも考えられ，実際，後述する評価実験においても実験でなければ長時間の使用に耐えることができないという回答を行った被験者が半数以上を占めた．さらに満足に使うことができない場合，そのツールは継続した使用がなされない場合があり，定期的かつ継続的に使用が求められるセキュリティツールにおいて致命的な問題となる．

表 1　ユーザビリティテストにおける実験環境とタスク
Table 1　Experimental environment and tasks.

クライアント OS	Windows 2000 Service Pack 4
セキュリティスキャナ	Nessus（インタフェース：評価対象）
発見される脆弱性の数	12
タスク 1	評価対象を用いてクライアント PC に脆弱性があることを診断する
タスク 2	Microsoft Update やパッチなどを用いて最低 1 つの脆弱性を修正する
タスク 3	すべての脆弱性を修正し，評価対象を用いて脆弱性がないことを確認する

3.3　本論文での評価環境と評価項目

表 1 は本研究のユーザビリティテストにおける実験環境とそのタスクを示したものである．フォーマットされた PC に Microsoft Windows 2000 をインストールし，それに Service Pack 4，Internet Explorer 6，セキュリティスキャナのみを導入した環境であり，4 章で予備調査を行った 2007 年 5 月 25 日〜6 月 6 日の時点では Nessus は脆弱性を 12 個発見する．

試験監督者は被験者に対しタスクのスタートとゴールに関する情報と，タスク実行に必要かつ最小限の情報のみを与えた．実験端末はインターネットに接続されており，実験中は操作方法や単語などを調べるために任意のサイトを閲覧することを許可されている．また，評価対象にはあらかじめ最低限の情報が初期値として入力してあり，もしほかに必要な情報がある場合は試験監督者に聞けば答えるという方式をとった．これらは被験者に対して初めから過剰に情報を与えすぎることを防ぐためである．なお，タスク制限時間はそれぞれ 40 分とし，制限時間を超えた場合はそのタスクは 40 分要したと記録し，試験監督者が次のタスクまでの作業を行った．

評価方法として ISO 9241-11 に着目したパフォーマンス測定を行った．パフォーマンス測定は実際に被験者が評価対象を操作して評価項目ごとに数値や文章で結果を得る測定方法であり，今回は ISO 9241-11 で示されているユーザビリティ3 要素である効果，効率，満足度について数値化して結果を得る．

効果はタスク達成率で計ることができる．もし被験者が独力でタスクを行えたのであればA，そうでなければB，タスク制限時間内にタスクを終えられなければCと評価される．Aとマークされた被験者数の割合が効果に関する評価結果となる．効率はタスク実行時間で示すことができる．どのタスクで多くの時間を費やしているかを記録する．なお，必要なスキャンに要した時間や，パッチなどのダウンロードやインストール，再起動に要した時間は省いてあるが，不必要な操作（たとえば不必要な診断リクエストや，IPアドレスの誤入力による診断の待ち時間など）を行った時間は加算している．効率に関する評価結果はそれぞれのタスク達成時間の平均時間から算出される．満足度は被験者の主観的な評価で行うことができる．満足度を評価するにあたり，ユーザビリティ専門の評価用紙である“Web Usability evaluation Scale”（WUS）[25]に基づいて評価を行った．WUSでは21項目の質問を行い，そこから生成される7つの評価因子でユーザビリティを評価する．その7つの評価因子とは，“操作の分かりやすさ”，“構成の分かりやすさ”，“見やすさ”，“反応の良さ”，“好感度”，“内容の信頼性”，“役立ち感”で構成される．本論文では被験者は5を最高点とする5段階評価を行い，満足度に関する評価結果は5段階評価の平均値から算出される．

4. 評価対象のセキュリティスキャナに対する予備調査

この章では評価対象のセキュリティスキャナ，すなわちNessusWXに対してパフォーマンス測定を行い，総括的評価を行う．**図1**にNessusWXのインタフェースを示す．この評価は本当にNessusWXのようなセキュリティツールに重大な問題点があるかどうかを見極めるための予備調査とし，被験者を6名とした．

総括的評価は3.2節で述べたユーザビリティ3要素を用いて，比較対象と開発したシステムに対して行われる．その2つの数値の結果を比べることにより，システムがどれだけ改善されたかを定量的に測定することができる．

4.1 コンテキストインタビュー

まず，ユーザ中心設計におけるステップ1と2として，ユーザのニーズを把握するために，被験者6名に対しコンテキストインタビューを行った．Holzblattらはコンテキスト調査法[24]を開発しており，この調査法を用いたコンテキストインタビューを行うことによって，ユーザはインタビュアに対して“教える”という意識が働き，通常のインタビューよりも掘り下げて聞くことができる．コンテキストインタビューにおいては被験者とのインタビューの中で質問内容を定めていくため，すべての結果を記述するのは長文となるため，いくつかの例を述べる．

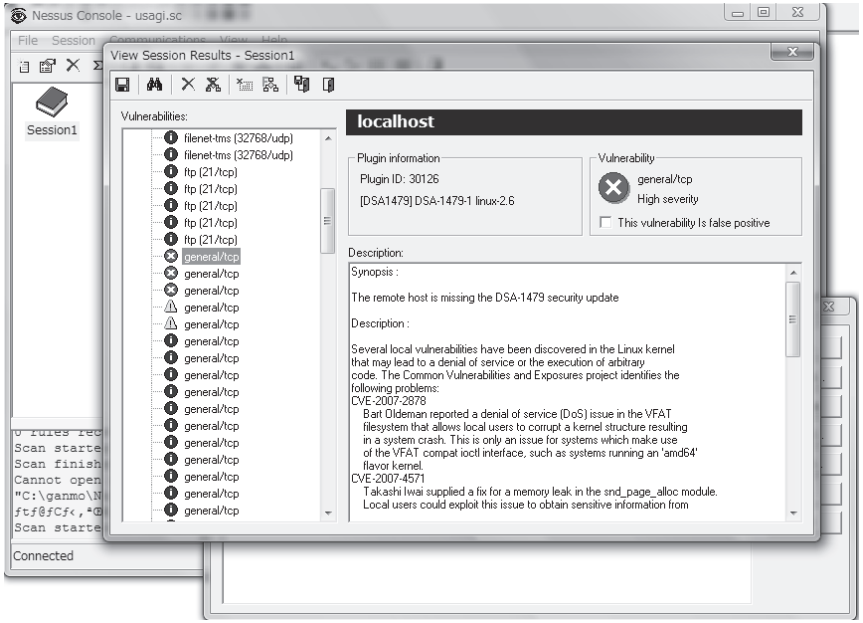

図1　NessusWXのスクリーンショット
Fig. 1　Screen shot of NessusWX.

被験者全員が自身のコンピュータを所持しているが，OSなどのアップデートは行っておらず，セキュリティの知識をさほど持ち合わせていない．またアップデートなどによって自身のコンピュータの動作が不安定になることや動作不能となることを恐れており，セキュリティ対策も能動的に行っているのではないことを把握した．そのほかにも多数の項目について把握することができている．

4.2 総括的評価（予備調査）

3.3節で示した実験環境とタスクと評価項目を用いて予備調査を行った．

4.2.1 効果に関する評価結果（予備調査）

すべての被験者はすべてのタスクを効果的に行うことができなかった．特にタスク1において，ほとんどの被験者はスキャンを受けることすらスムーズに行えなかった．さらに，3名の被験者は間違ったIPアドレスを入力した．これはユーザの判断能力を超えた誤った行動を容易に行えてしまうということであり，システムとして深刻な問題であるといえる．

4.2.2　効率に関する評価結果（予備調査）

すべての被験者はタスク 1 に多くの時間を要しており，NessusWX をスムーズに扱うことができなかった．また，ほとんどのユーザがすべてのタスクを完了するのに 40 分以上かかった．

4.2.3　満足度に関する評価結果（予備調査）

5 段階評価の中間点である 3 より低い評価，つまりネガティブな評価を付けた被験者が多かった．全体的に数値が低く，特に “構成の分かりやすさ”，“反応の良さ”，“役立ち感” の項目において低い評価であった．情報量の少なさや，逆にあふれかえる量の情報に困惑し，低い評価を行った被験者がほとんどであった．特にすべての被験者は “分からない単語が多数ある” 点に不満を持っていた．

以上の結果から，NessusWX には何かしら重大な問題点があると判断し，具体的な問題把握と修正作業を行うこととした．

5.　形成的評価

次に，具体的に比較対象がどのような問題点を持っているか把握するため 6 名の被験者を用いて形成的評価を行った．具体的な実験環境とタスクは 3.3 節で用いたものと同じである．形成的評価は開発の途中段階で幾度も行われ，開発者はユーザビリティ評価のフィードバックからユーザの多くの意見を把握し，採用する．

本論文では形成的評価として，Lewis らによって開発された “思考発話法” を用いたユーザテストをユーザ中心設計におけるステップ 3 として行い，Nielsen によって提唱された 10 ヒューリスティックス[27] に基づいて評価を行うヒューリスティック評価をユーザ中心設計におけるステップ 4 と 5 として行う．

5.1　思考発話法

思考発話法はユーザが操作中に考えたことや感じたことをすべて口に出す方法であり，そのシステムに対して被験者がどの点に関して困難を感じるかを具体的に把握することができる．しかし被験者がすべてを口に出さない場合も考えられるため，試験監督者は被験者のディスプレイや視線の動き，キーボードやマウスなどといったすべての言動を記録し解析することとした．

表 1 で示した実験環境においてタスクを，被験者 6 名を用いて，思考発話法による形成的評価を行った．

5.1.1　思考発話法で発見されたタスク 1 での問題点

被験者は NessusWX を起動し，Nessus のデーモンが起動しているサーバとの接続を行う．しかし NessusWX は簡単なメッセージと分かりにくいアイコンとメニューで構成されたインタフェースであるため，すべての被験者は NessusWX を起動した時点で戸惑いがみられた．被験者は次に何をすればよいかをまったく把握できない状態に陥り，NessusWX の操作方法を解説するサイトを探す作業を行っていた．また，被験者が利用している端末の IP アドレスを入力するウィンドウが表示された場合，試験監督者に IP アドレスの調べ方を聞けば IP アドレスを教えるという方式をとっていたが，被験者 3 名は解説サイトに掲載されていたアドレスをそのまま入力してしまうなど，間違った数値を入力した．実環境で同様の作業を行った場合，意図しない端末に対して診断を行ってしまう可能性があり，致命的な問題点としてあげられる．

5.1.2　思考発話法で発見されたタスク 2 での問題点

タスク 2 において，結果表示がたとえ母国語であっても，すべての被験者は専門用語を理解することができなかった．また，結果表示は平板であり，多くの情報が羅列されているだけであるため，彼らは脆弱性の対処法を理解するのに多くの時間を費やしていた．一般的にセキュリティスキャナは最も目立つ最初の箇所に脆弱性の説明文を表示する．しかしそれらの情報はセキュリティに関する知識がなければまったく理解できない文章であり，今回用いた被験者はそれらを理解しようとして多くの時間を割いた．これは効率の面において問題である．また，今回の場合，ほとんどの対処法は Microsoft のサイトのアドレスを指し示し，そこのパッチを適用するように表示されている．しかしその Web サイトは情報を読み取ることが難解であり，被験者にとって理解することが困難である情報提供がなされているため，必要な知識を持ち合わせていないすべての被験者は必要な情報が読み取れず，正しいパッチをダウンロードすることや，他の適切な方法を理解することができなかった．

5.1.3　思考発話法で発見されたタスク 3 での問題点

タスク 3 において，監督者はタスク 2 においてパッチを適用した被験者に対して，残り 11 個の脆弱性を修正するのには多くの時間を要するため，ほかに方法はないかと提案した．たとえば Microsoft Update を使用した場合，数回実行するだけで必要なパッチがスムーズに適用され，ユーザが適用を望まないパッチがある場合はそれを除いて更新を行うことができる．しかし，その情報がページの分かりにくい箇所に記述されており，また，被験者の目に入っても利便性などが伝わらなかったため，それを利用するに至るまでに多くの時間を費やした被験者が 5 名存在した．また，つねに 1 回の実行ですべてのパッチ適用やアップ

表 2　10 ヒューリスティックス
Table 2　10 heuristics.

1	Visibility of system status
2	Match between system and the real world
3	User control and freedom
4	Consistency and standards
5	Error prevention
6	Recognition rather than recall
7	Flexibility and efficiency of use
8	Aesthetic and minimalist design
9	Help users recognize, diagnose, and recover from errors
10	Help and documentation

デートが行われるわけではなく，そうでない場合は再度 Microsoft Update を行わなければならない．たとえば新しいサービスパックが提供される場合，Microsoft Update はサービスパックのアップデートに関する情報のみを表示し，他のパッチに関する情報は表示しない．もし 1 回だけ Microsoft Update を行った場合，サービスパックを導入しただけであり，他のパッチなどはインストールされていない．したがって，Microsoft Update はほかのすべてのパッチが表示されるまで繰返し実行する必要がある．しかしその記述が Microsoft のサイト上には掲載されていないため，1 回のみアップデートを行い，脆弱性はすべて修正されたと信じたすべての被験者はパッチが適用されていない状態で再スキャンを行った．今回の場合は監督者がそれはゴール地点ではないことを注意することができたが，実環境においては，必要回数の Microsoft Update が行われず，必要なコンポーネントが更新されない致命的な状態のままで放置される恐れがある．

以上から，NessusWX は操作面で困難性をかかえているだけでなく，結果表示も情報を正確に読み取ることが困難である Microsoft のサイトを指し示すだけであり，ユーザブルでないことが確認できた．この調査によって，主にユーザが潜在的に戸惑いを示すポイント，たとえば操作に関する説明をヘルプなどを読まない限り分からない点はユーザにとって手間であるというより不快に感じている点であるなど，通常の調査では把握できない部分まで把握することができた．そのほかにも多くの問題点が発見され，それらの問題点を修正したプロトタイプを作成した．

5.2　10 ヒューリスティックスおよびヒューリスティック評価

10 ヒューリスティックスにおいて，Nielsen は多くのユーザビリティの問題を分析し，それらの問題の背後に潜在するユーザビリティの 10 個の原則を抽出した．**表 2** に 10 ヒューリスティックスを示す．この 10 ヒューリスティックスに基づき，Nielsen によって提案されているユーザビリティインスペクション[28] に基づいたヒューリスティック評価を行うことによって，提案するセキュリティスキャナの問題点の発見と改善を行う．

プロトタイプのセキュリティスキャナに対し，3 名の被験者を用いて 4 回のヒューリスティック評価を行った．プロトタイプを実際に動作してもらい，10 ヒューリスティックスの観点から問題点を見つけ出すヒューリスティック評価を繰り返し行った．この評価において，10 ヒューリスティックスで指摘されているユーザビリティの問題点を洗い出すことができている．この評価で得られた代表的な解決策の例として，“基本的な情報は最初から初期設定してあるべきである”，“付加された情報は最初の段階では非表示とし，初期段階では必要最低限のみを表示する” といった点があげられる．開発するシステムでは脆弱性情報の表示画面において，初期の段階では脆弱性の概要は表示せず，対処法を特に目立つ形で表示し，ユーザの要求があったときに要求された項目を表示する仕組みをとった．5.1 節の評価とあわせ，合計 24 個の大きな問題点が発見され，修正された．

5.3　提案するセキュリティスキャナの構成

提案するシステムはセキュリティスキャナとインタフェース，そしてサポートデータベースから成り立つ．それぞれの部分は前述の評価結果を反映させて実現されている．

5.3.1　システムの核となる部分

提案するシステムは Nessus をセキュリティスキャナの核として採用し，新たにインタフェースの開発を行った．前述のとおり，スキャンスクリプトの生成や動作チェックや管理は莫大なコストがかかり，そもそもいくつかのオープンコミュニティがある現在において別のセキュリティスキャナを 1 から作り直すことは車輪の再発明であると判断したためである．インタフェースが Nessus と直接通信を行い，Nessus をそのまま動作させることによって，提案するセキュリティスキャナは信頼性をそのまま引き継ぐことができる．

5.3.2　インタフェース

本論文で実現するセキュリティスキャナシステムは Web ベースのセキュリティスキャナであり，インタフェースは CGI を介してセキュリティスキャナと通信を行う．すなわち，セキュリティスキャナのクライアントアプリケーションのインストール作業をいっさい必要としない．インタフェースにはグラフィカルで柔軟なインタフェース設計が容易に可能である Adobe Flash を採用した．Adobe Flash はインターネットに接続されているほとんどすべてのクライアントにおいて動作可能であり，ほとんどのケースにおいて Adobe Flash は事前にインストールされている．もしインストールされていない場合においても，それのイン

ストール作業は容易である．このことから提案するセキュリティスキャナシステムはOSに依存しないといっても過言ではなく，Adobe Flashが動作するすべてのOSとブラウザとそれらのバージョンの組合せにおいて同一の動作が行われるため，コストの削減にもなる．

このインタフェースは初期状態では必要最小限のインタフェース（ターゲットIPアドレスを入力するテキストフォーム，次の画面へ遷移するボタン，拡張機能へ遷移するボタン）のみを提供する．拡張機能へ遷移するボタンを選択することによって，ユーザは詳細なオプション設定を行うことができる仕組みをとっている．このインタフェースの詳細なオプション設定はNessusの機能と1対1で対応しているため，Nessusの機能性を何ひとつ失っていない．これは対象物のユーザビリティを向上させても，機能性を保持している，すなわち両立させていることを示している．

さらにこのセキュリティスキャナはマウスクリックのみで操作が可能であり，ユーザの判断能力を超えた操作を行えないようにする簡単な仕組みをとり入れている．たとえば，もしユーザが，表示されているあるテキストを理解できない場合，そのテキストにマウスカーソルをあわせると詳細な情報の表示を行う仕組みをとっている．また，このインタフェースはスキャン状況を動的に表示する．

さらに，このインタフェースは言語ごとのメッセージを含めたXMLファイルを読み込むことによる表示言語の拡張性を持ち合わせている．現在は英語と日本語の拡張表示オプションを持ち合わせているが，他言語においてもXMLファイルを追加することによって容易に拡張可能である．

5.3.3　ユーザサポートデータベース

現在配布されているセキュリティスキャナは結果表示として脆弱性の説明文と解決法を記述している．しかし5.1節の形成的評価の結果より，そのような情報はユーザを惑わすだけであることを把握している．なぜならCERT/CC，CVE，SecurityFocus，JVN，OSVDB[29]などが提供している脆弱性情報はほとんどのユーザにとって膨大で平板であるか，かなり難解であるかいずれかであるためである．

そこでユーザをサポートするためのサポートデータベースを開発することとした．事前にサポートデータベースは前述の脆弱性情報提供ベンダやそれが参照するサイトなどから自動的に，XML形式で配布されていれば収集・解析を行い，HTMLやTXT形式であれば収集した後に可能な限り情報の解析（たとえば特定ベンダが一定フォーマットによって提供されているのであればその特徴を用いて解析を行う．多くのベンダはそれぞれ一定のフォーマットで情報提供を行っている）を行い格納する．たとえばMicrosoft社のソフトウェアやOSに存在する脆弱性はMicrosoft Updateによって修復可能であり，サポートデータベースはユーザに対してMicrosoft Updateの使用を勧め，それの使い方も簡素に説明する．もしMicrosoft Updateによる脆弱性対処が不可能である場合はパッチを当てることを勧め，サイトの使い方を説明する．

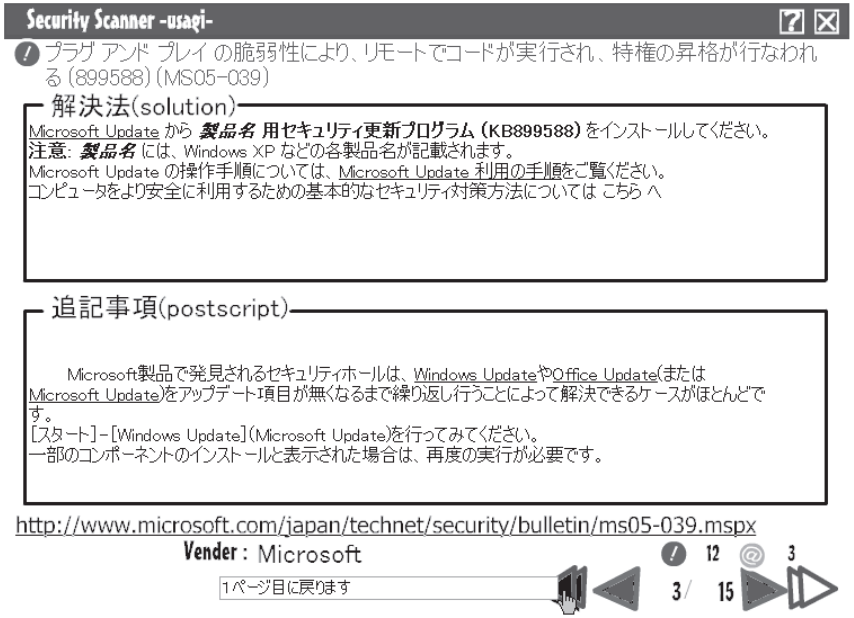

図2　開発したシステムのスクリーンショット
Fig. 2　Screen shot of our developed security scanner system.

図2は開発したシステムの診断結果表示画面のスクリーンショットを示す．先に述べたとおり，初期状態において診断結果表示画面では対処法を特に目立つ形で表示し，あわせてユーザサポートデータベースによる追記事項を表示している．今回のケースであればMicrosoft製品の脆弱性情報であるため，Microsoft Updateの使用について述べられている．また，ボタンにマウスカーソルをあわせたときにインタフェース内にポップアップメッセージが表示され，ユーザの操作に対する補助が行われている．

6. 比較対象と提案するシステムの総括的評価と意識調査

最後に比較対象である NessusWX と 5 章で開発を行ったシステム（以下，提案システム）に対する総括的評価を行った．

6.1 総括的評価

総括的評価をパソコンを使い慣れているユーザから必要に応じて使うユーザの中から，無作為に選択した 30 名の被験者を用いて 3.3 節で述べたパフォーマンス測定の手法によって行った．実験環境とタスクと評価項目は 3.3 節で述べたとおりである．なお，実際の実験では比較対象と提案システムは被験者ごとにランダムな順番で行っている．

6.1.1 効果に関する評価結果

効果に関する評価結果を**図 3** に示す．ほとんどの被験者が提案システムにおいてスムーズにタスクを行うことができた．サーバへの接続は Web ブラウザに URL を入力するだけで完了し，ターゲット IP は初期値としてアクセスを行った IP アドレスが入力されている．提案システムは被験者に対し判断力を超えたスキャンを行わないために簡単なチェックリストの画面を表示する．また，操作はすべてマウスクリックを数回行うだけで完了可能であり，インタフェース内のバルーンヘルプに従って操作をしていくことにより操作が完了できる．これらにより，すべての被験者はタスク 1 をスムーズに操作を行うことができた．

6.1.2 効率に関する評価結果

次に，効率に関する評価結果を**図 4** に示す．NessusWX は平均して 56 分 4 秒要しているのに対し，提案システムは 19 分 6 秒ですべてのタスクを完了している．すべての被験者は NessusWX と比べて短い時間でタスクを終えることができた．特にタスク 1 において，すべての被験者は大幅な時間短縮を行うことができている．このことから，被験者は NessusWX よりも効率的にタスクを行えたことは明白である．これはユーザに対し初期状態では余分な情報提供は行わず，必要最小限のインタフェース提供を行ったことに起因していると考えられる．

6.1.3 満足度に関する評価結果

次に，満足度に関する評価結果を**図 5** に示す．すべての被験者は NessusWX より提案シ

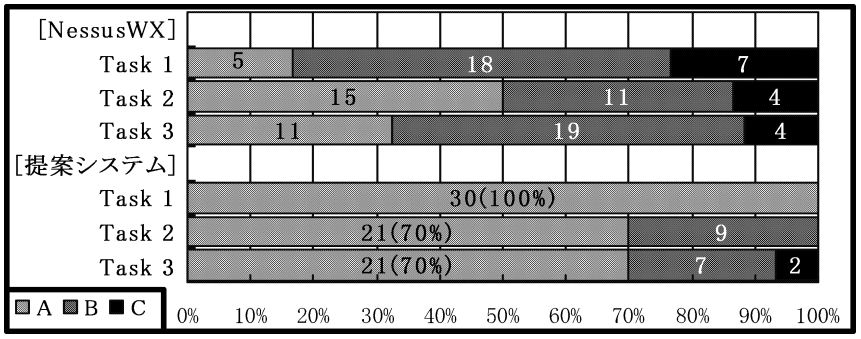

図 3　パフォーマンス測定の結果（効果）
Fig. 3　Result of performance measurement (effectiveness).

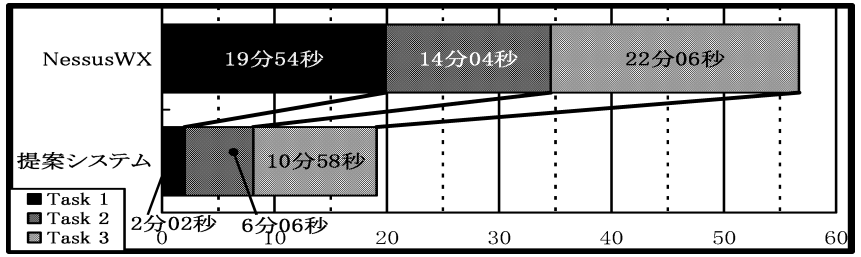

図 4　パフォーマンス測定の結果（効率）
Fig. 4　Result of performance measurement (efficiency).

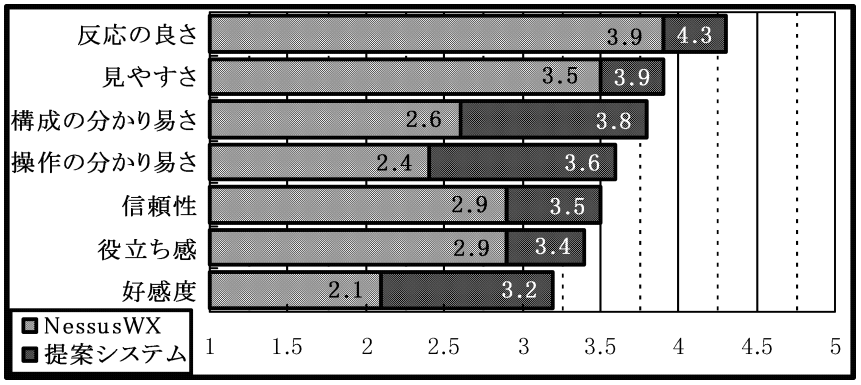

図 5　パフォーマンス測定の結果（満足度）
Fig. 5　Result of performance measurement (satisfaction).

ステムの方が高い数値をマークしている．特に NessusWX において評価が低かった “構成の分かりやすさ”，“操作の分かりやすさ”，“好感度” において大幅な差があり，大きく改善できているといえる．また，WUS の 7 つの評価因子中の “内容の信頼性”，すなわち，使用者がそのインタフェースに対して内容が使用者にとって信用できる内容であるかどうか，また文章表現が適切であるかどうかについて，情報量が限られているにもかかわらず NessusWX よりも良い結果が出ていることは，セキュリティスキャナはユーザにとって必要最小限の情報提供を行うだけで十分であるということを指し示している．

6.2　母国語表示による差の調査

提案システムは言語表示に関する拡張機能を持ち合わせているため，調査実験において母国語による表示を行っているが，母国語表示であるゆえに優位な結果が出ているのではないかという可能性がある．この点に関して追加実験を行った．この調査には “ペーパープロトタイピング” [30] を用いた．ペーパープロトタイピングは紙の上にアプリケーションを再現してユーザビリティテストを行う評価法であり，プロトタイプのインタフェースに対するユーザビリティ評価を行う場合，紙上のインタフェースに対して評価を行うだけで十分な結果を得ることができることが知られている．今回の場合は NessusWX の日本語インタフェースと提案システムをそれぞれ紙に再現し，ユーザビリティテストを行った．なお，被験者数は 20 名である．

効果に関する評価結果については，提案システムにおいて A をマークした被験者は 19 名であったが，NessusWX は 1 名のみであった．さらに，NessusWX において C をマークした被験者はタスク 1 において 13 名と，診断を受けることすら行えなかったことが分かる．また，NessusWX を操作するにあたり，母国語で書かれているにもかかわらずインターネットで操作方法を閲覧した被験者が 16 名おり，間違った IP アドレスを入力した被験者も 3 名いた．

効率に関する評価結果については，提案システムは全被験者平均が 4 分 43 秒であったが，NessusWX は 21 分 31 秒であった．特に NessusWX はタスク 1 において 12 分 34 秒，時間がかかっている．

満足度に関する評価結果については，提案システムは 4.00 ポイントであったのに対し，NessusWX は 2.62 ポイントであった．NessusWX は 6.1.3 項での評価で数値が低かった項目は依然低く，表示言語に関係なく満足度は低い結果となった．

これらのことは，母国語表示をサポートするだけではユーザビリティの改善にはつながらないことを示しており，致命的な問題点はほかにあることを示唆している．5.1 節や 5.2 節で行ったような形成的評価のプロセスを行わない限り潜在的な問題点の発見と改善にはつながらないことがこの実験によって示されている．

6.3　セキュリティに関する意識変化の調査

6.1 節の総括的評価に参加した 30 名に，今回の評価実験は核となるシステムは同一（つまり，被験者にとってどちらのシステムも得られる結果は同一であった）という趣旨を説明したうえで，選択項目と自由回答を組み合わせた形式で意識調査を実施した．NessusWX と提案システムではどちらの方が使いやすいかという設問（NessusWX の方が使いやすい，どちらともいえない，提案システムの方が使いやすい，分からないの選択肢から 1 つ選択）に対し，30 名中 28 名が，提案システムの方が使いやすいと回答したうえで，“このように簡素化されたシステムであれば今後も使いたい” といった興味を示す回答や，“安全性が増す” といった意見や “こういったシステムは啓発になる” といった意見が見受けられた．また，提案システムを利用してセキュリティに対する意識に変化はあったか？という設問（大きく変わった，変わった，どちらともいえない，変わらない，まったく変わらない，分からないの選択肢から 1 つ選択）に対し，30 名中 22 名がセキュリティに対する意識が変わったと回答しており，“便利さだけでなく危険性という視点を持つことができた” といったきっかけを実感した被験者や，“難しそうであるという印象があったが今回の実験で印象が変わった” と回答した被験者の意見が見受けられた．また，調査時にセキュリティ対策をまったく行っていない被験者 10 名に対し，提案システムのようなセキュリティ製品であれば使用したいか？という設問（積極的に使いたい，使用したい，どちらでもない，使用したくない，まったく使用したくない，分からないの選択肢から 1 つ選択）に対し，10 名中 9 名が “積極的に使いたい” と回答しており，そのうち 1 名がこのように容易に使用可能かつ効果的であれば積極的に使いたいと回答している．このようなユーザビリティと機能性を両立させたシステムの普及はユーザの意識向上につながることが分かった．

7. ま と め

本論文では，既存のセキュリティスキャナに対しユーザビリティ評価を行い，すべてのユーザが効果的にそれらを使えないことを発見した．そこでユーザ中心設計をとり入れ，開発指標としてユーザの意見を随時とり入れながらインタフェース開発を行い，重大なユーザビリティ問題点の発見と改善を繰り返し行った．さらにそれが ISO 9241-11 のユーザビリティ3 要素を満たすものであることを評価実験により示した．開発したシステムを用いた場合，ユーザは効果的にそれを使うことができるようになり，ユーザブルでなかった有用なセ

キュリティツールが使えるようになるだけでなく，効果的に効率良く満足に使えるシステムが実現された．また本システムの普及は，現在の一般ユーザがかかえる大きな課題である，セキュリティに対する意識向上につながることを示した．

今後の課題として，セキュリティ製品全般においてユーザビリティとセキュリティを両立させる開発手法の確立を行うことがあげられる．また，ユーザが必要とする情報の提供方法においても，工夫次第でユーザビリティ向上が見込めるため，今後の課題としたい．

参 考 文 献

1) 情報処理推進機構：2007年コンピュータ不正アクセスの届出状況について（オンライン）．http://www.ipa.go.jp/security/txt/2008/documents/2007all-cra.pdf (参照 2009-02-10)
2) 情報処理推進機構：情報セキュリティに関する脅威に対する意識調査（2007年度第2回）（オンライン）．http://www.ipa.go.jp/security/fy19/reports/ishiki02/ (参照 2009-02-10)
3) 情報処理推進機構：情報セキュリティに関する脅威に対する意識調査（2007年度第1回）（オンライン）．http://www.ipa.go.jp/security/fy19/reports/ishiki01/（参照 2009/02/10)
4) Furnell, M.S., Jusoh, A. and Katsabas, D.: The Challenges of Understanding and Using Security: A Survey of End–users, *Computers & Securities*, Vol.25, pp.27–35 (2006).
5) Cranor, F.L. and Garfinkel, S.: *Security and Usability*, p.xi, Oreilly & Associates Inc. (2005).
6) Norman, A.D. and Draper,W.S.: *User Centered System Design*, Lawrence Erlbaum Assoc Inc, Hillsdale, NJ (1986).
7) International Organization for Standardization: ISO 9241-11:1998, Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability (1998).
8) JPCERT/CC and IPA：JVN#66077895：ウイルスセキュリティおよびウイルスセキュリティZEROにおけるサービス運用妨害（DoS）の脆弱性（オンライン）．http://jvn.jp/jp/JVN66077895/index.html (参照 2009/03/08)
9) National Institute of Standards and Technology: National Vulnerability Database (NVD) National Vulnerability Database (CVE-2008-4429), National Vulnerability Database (online). http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2008-4429 (accessed 2009-03-08)
10) Anonymous: *Maximum Security fourth edition*, pp.221–230, Sams Publishing, Indiana (2002).
11) Tenable Network Security: Tenable Network Security (online). http://www.nessus.org/nessus/ (accessed 2009-02-10)
12) National Institute of Standards and Technology: National Vulnerability Database (NVD) National Vulnerability Database (CVE-2007-3215), National Vulnerability Database (online). http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2007-3215 (accessed 2009-02-10)
13) 毛利公美，曽根直人，高橋秀郎，神薗雅紀，森井　昌：ネットワークサーバにおける脆弱性自動監査システム，コンピュータセキュリティシンポジウム 2003，No.4B-2, pp.271–276 (2003).
14) JPCERT/CC and IPA：Japan Vulnerability Notes（オンライン）．http://jvn.jp/ (参照 2009-02-10)
15) 藤平俊行，大岸伸之，森尻智昭，才所敏明：脆弱性情報を用いた分析手法に関する検討，2003年暗号と情報セキュリティシンポジウム，No.7A-3, pp.481–486 (2003).
16) 小手川祐樹，田端利宏，堀　良彰，櫻井幸一：モバイルエージェントによるエンドホスト情報収集管理方式の提案，2005年暗号と情報セキュリティシンポジウム，No.4B1-2, pp.1717–1722 (2005).
17) National Cyber Security Division of the U.S. Department of Homeland Security: OVAL - Open Vulnerability and Assessment Language (online). http://oval.mitre.org/ (accessed 2009-02-10)
18) The Inprotect Team: Inprotect – Web based front end for Nessus and Nmap (online). http://inprotect.sourceforge.net/ (accessed 2009-02-10)
19) 早坂明哲，木村達洋，瀬川典久，宮崎正俊，山崎清之，村山優子：ソフトウェア操作における待ち時間が操作者の心理生理学的状態に及ぼす影響，電学論 C，Vol.127, No.10, pp.1770–1779 (2007).
20) Yoshimoto, M., Bista, B.B. and Takata, T.: Development of security scanner with high usability, *Proc. 18th International Conference on Advanced Information Networking and Applications*, Vol.1, pp.139–144, IEEE Computer Society and Communications Research Laboratory, Japan (2004).
21) Yoshimoto, M., Katoh, T., Bista, B.B. and Takata, T.: Development and Evaluation of New User Interface for Security Scanner with Usability in Human Interface Study, *LNCS*, Vol.4658, pp.127–136, Springer (2007).
22) Nielsen, J. and Landauer, K.T.: A mathematical model of the finding of usability problems, *Proc. INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, pp.206–213 (Apr. 1993).
23) Braz, C., Seffah, A. and Raihi, M'.D.: Designing a Trade-Off Between Usability and Security: A Metrics Based-Model, *LNCS*, Vol.4663, pp.114–126 (2007).
24) Beyer, H. and Holtzblatt, K.: *Contextual Design*, Morgan Kaufmann Pub., San Fransisco, CA (Dec. 1999).

25) 仲川　薫，須田　亨，善方日出夫，松本啓太：ウェブサイトユーザビリティアンケート評価手法の開発，第 10 回ヒューマンインタフェース学会紀要，No.3114, pp.421–424 (2001).
26) Lewis, C.: Using the ‘thinking-aloud’ method in cognitive interface design, *IBM Research Report*, RC 9265 (1982).
27) Nielsen, J.: Enhancing the explanatory power of usability heuristics, *Proc. SIGCHI Conference on Human Factors in Computing Systems: Celebrating Interdependence*, pp.152–158, Boston, MA (1994).
28) Nielsen, J.: *Usability Inspection Method*, John Wiley & Sons Inc., Hoboken, NJ (1994).
29) Open Source Vulnerability Database (OSVDB): OSVDB: The Open Source Vulnerability Database (online). http://osvdb.org/ (accessed 2009-02-10)
30) Snyder, C.: *Paper Prototyping*, Morgan Kaufmann Pub., San Fransisco, CA (2003).
31) Nielsen, J.: *Usability Engineering*, Morgan Kaufmann Pub., San Fransisco, CA (1994).
32) 樽本徹也：ユーザビリティエンジニアリング，オーム社，東京 (2005).

(平成 21 年 4 月 30 日受付)
(平成 21 年 10 月 2 日採録)

吉本　道隆（正会員）

1980 年生．2003 年岩手県立大学ソフトウェア情報学部ソフトウェア情報学科卒業．2005 年同大学大学院ソフトウェア情報学研究科修了．現在，清泉女学院大学助教．セキュリティおよびユーザビリティ工学，HCI に関する研究に従事．電子情報通信学会，ヒューマンインタフェース学会各会員．

加藤　貴司

1971 年生．2001 年東北大学大学院情報科学研究科博士後期課程修了．現在，岩手県立大学ソフトウェア情報学研究科講師．博士（情報科学）．マルチエージェントシステムにおけるエージェントの協調に関する研究に従事．人工知能学会，電子情報通信学会各会員．

ベッド バハドゥール ビスタ（正会員）

1967 年生．1991 年 York 大学電子工学科卒業．1997 年東北大学大学院情報科学研究科博士課程修了．1997 年から 1998 年宮城大学勤務．現在，岩手県立大学ソフトウェア情報学研究科助教授．博士（情報科学）．プロトコルの仕様記述と合成，モバイル通信に関する研究に従事．

高田　豊雄（正会員）

1962 年生．1989 年大阪大学大学院基礎工学研究科博士後期課程修了．現在，岩手県立大学ソフトウェア情報学研究科教授．工学博士．セキュリティと誤り制御通信に関する研究に従事．電子情報通信学会，情報理論とその応用学会，IEEE，ACM 各会員．