資料２

技術戦略専門委員会グランドチャレンジ検討ワーキンググループ
第１回会合議事要旨（案）

１．日時　平成２０年８月２８日（木）１５：００～１７：４０

２．場所　内閣府本府５階特別会議室

３．出席者

［主査］
後藤　滋樹（早稲田大学教授）
［主査代理］
安達　　淳（国立情報学研究所教授）
［委員］
伊藤　光恭（NTT 情報流通プラットフォーム研究所　セキュアコミュニケーション基盤プロジェクト　グループリーダ）
加藤　雅彦（株式会社　アイアイジェイテクノロジー　IBPS 本部　システム技術部　部長代理）
楠　　正憲（マイクロソフト株式会社 CTO 補佐）
西本　逸郎（株式会社ラック　サイバーリスク総合研究所　所長）
二木　真明（住商情報システム株式会社　情報セキュリティ・IT 統括部　担当部長）
松並　　勝（ソニーデジタルネットワークアプリケーションズ株式会社　セキュリティテクノロジマネージャ）
三河尻浩泰（株式会社富士通大分ソフトウェアラボラトリ　セキュリティセンター長）
森山　浩幹（株式会社エヌ・ティ・ティ・ドコモ　法人事業部　ソリューションビジネス部　担当部長）
山田　安秀（情報処理推進機構　セキュリティセンター長）
（五十音順）
［政府］
内閣官房情報セキュリティセンター内閣参事官
内閣官房情報セキュリティセンター情報セキュリティ補佐官
内閣府政策統括官（科学技術政策・イノベーション担当）付参事官
警察庁情報通信局情報技術解析課課長補佐（代理出席）
総務省情報セキュリティ対策室課長補佐
文部科学省大臣官房政策課情報化推進室長

経済産業省商務情報政策局情報セキュリティ政策室課長補佐

4．議事概要

○　なぜグランドチャレンジをやるのかというところを説明したいが、長期的な計画を立てて究極の目標を達成していくという、長期型の研究プロジェクトは政府が行う研究投資の一つの形である。多くのそうした研究開発は 5 年レンジが多い。その先を考えた時に今の研究プロジェクトの進め方は構造的な問題があるためである。

どこに問題があるかと考えたときに、現在研究開発に政府系のお金を使おうとすると、目標を立てた後に要素還元的にサブテーマに分け、進捗評価である中間評価と、終了後に事後評価が行われて成果利用計画がなされる。走り始めたら計画通りに進めるので、要素還元的に片付くものは良いが、周りのリスク要因が変わったので計画を変更しようとしてもまず不可能で最後まで走らなければならないし、計画が早く終了したので先に届くようにしようとすると計画書に書いていないからダメだと言うことになる。

短期だとこれでも問題ないが、3 年程度でも IT のジャンルでは研究対象が製品化された製品が出てきて研究価値が無くなるような不幸な事態も発生しうる。これが 5 年～10 年になるとなおさら今の管理体制では無理である。

そうしたことを考慮すると、他のポイントからプロジェクトを管理すべきで、成果は要素還元的なものよりも、社会的な視点からの成果目標とか、成果活用を最大化するような計画を作っておいて、成果プロセスをどう活かそうかを考えるチームを作成したり、自分たちのチームの中でリファクタリングをしながら方向を変えていく方法や、マーケットを見ながら研究の方向性があわなくなってきたときには止めたり、新しく組み込んだりと言うことをするべきだ。

但し、これは政府の枠組みの中では非常に行いにくい。まず、目標が大まかになるため財務省から予算を確保しにくくなるし、途中で組み替えるため、会計検査院の検査も通りにくい。

それでも、情報セキュリティにおいては、このような仕組みを構築する必要がある。なぜならば、常に動的に目標は変化するし、守る技術は攻撃側と比較して、攻撃する側がフリーハンドで攻撃できるのに対して、防御側は法律等に手足を縛られるという攻撃側有利な非対称性がある。そのため、硬直的な研究開発体制では、セキュリティを達成できない。加えて、情報を保護するという究極の目標はあまり変わっていないし、社会的な要請は非常に強くなってきている。プライバシー・コンシャス・コンピューティングなどではそれが顕著である。

グランドチャレンジにおいては、そういった面も考えながらできればいいと考えている。

○　グランドチャレンジについては、ここ 3 年ぐらい係わっており、昨年は第 1 次案を作成した。グランドチャレンジとは何かと言うところから始まった場で、ビジョナリーなゴールと言ったため、イメージだけが先行して話の粒度がそろっていないものがある。

しかし、それでは政策として一体感のある研究開発はできないので、今年度の WG は、政府としての将来の社会ビジョンを示してそれを実現するための技術を議論しようと考えている。

対象分野は政府として関心のある公共性の高い基盤システムのサービスを主にやっていきたいと思っている。もちろん、電子政府を例にとると電子化された政府の部分だけでなく、それを使うユーザの PC に代表されるように面を貼ったような大きなシステムが電子政府となるので、幅広くやっていきたい。

グランドチャレンジとしてビジョンを作っていくと言うときに、3 年・5 年・10 年を想定しており、特に、3 年後を見た技術開発と 10 年後の技術開発は取組方法が大きく異なっている。3 年後の場合は現在の技術の演繹でシーズ指向によって予測できる。10 年後の技術の場合は、サービスを予測して、それが行き渡った社会を考えてそれに必要なものを考えるニーズ指向で考えると言うことだと思う。

本 WG においては、どちらでやるかを決めずに、両方から検討していきたい。

検討する視点しては、領域と粒度というものを例として示してみた。また、フェーズ分けも補足の資料が存在する。これにとらわれず、検討してほしい。

○　自己紹介をかねて、コメントをお願いしたい。

○　現在情報爆発というキーワードで研究をしている。研究においては現実的な問題をこなしているが、現実は絵に描いたようには動かないと実感している。10 年後となると見識を問われているので、厳しくなるとは思うが、精一杯がんばりたい。

○　現在は、ネットワークセキュリティ関係の研究をしていてボットネット、VoIP のセキュリティ、Web のセキュリティに取り組んでいる。10 年後のセキュリティという話題が出たので、将来予測を考えていたが、攻撃者に先回りして、攻撃者が何を意図しているかを考えた技術が必要だと思われる。

今想定しているのはマルウェア解析・捕獲が攻撃者の意図を予測するために重要だと思っているので、今取り組んでいるものである。

○　企業の情報システム、システムインテグレーション、脆弱性検査、フォレンジック等に色々係わってきた中で、セキュリティは細かく・狭く、ターゲットがピンポイン

トになりがちだと思っていた。JNSA で情報セキュリティの 3 年後を考えたときは、あまり明るい未来を見いだせなかったので、夢のある将来を描けると良いと思っている。

○　ここ５～６年でセキュリティトレンドには大きな変化があった。2003 年に発生した Blaster というのは最後の愉快犯的な事案であった。その後ウイルス感染が減っているのではないが、それまでのように騒ぎにはならず、情報を盗んだり、ボットネットで SPAM を発信するなどビジネスとして行われるようになってきている。そのため、消費者からは不可解になってきている。

もう一つの大きな変化は、特にフィッシング以降強く感じるが、ソフトウェアの脆弱性だけでなく、使われ方如何で人々は被害を受けると言う事例が非常に増えている。プライバシーに関しても元々OS では情報が漏れないようにするかという論点であったが、今も、google ストリートビューが問題になっているが、今後クラウドコンピューティングやライフログの様に新しい技術が出てくる中で、もっと幅を広げて考えていく必要がある。今の技術の延長のみならず、10 年後のライフスタイルとセットで考える必要があると思っている。

○　ベンチャー系のセキュリティ製品を販売するべく技術評価等をやっているが、仕事の中で得たものというのは、何がどこで売れるかわからないと言うことである。先ほどの話にもあったように、研究しているといきなり新しい製品が出ると言うことも経験しているので、そのような経験も生かしていきたいと思っている。

今は、自社のセキュリティの技術面からの企画で、管理の仕組みという業務もやっているので、そういうことを活かして議論に係わって行きたい。

○　最近は、サイバーなスペースのリスク研究と言うことで、緊急時対応サービスを通じてフィールドで研究開発をしているとも言える。その中から見えることだが、半年ほど前はカード情報を盗むときにどの情報を盗っていったかがわかるようになっていた。しかし、犯罪組織が自らの利益を守るために、現在ではわからないように盗っていくようになってきている。このように半年の間でも手口が変わる。

そうなると、レーダーの能力、見抜く能力が重要になってきている。

また、こういうセキュリティ分野の技術は海外に頼っていることが多いが、各分野で国産の自給率も計画的に保つ必要があるのではないかと思う。

最後に、分野という話から言うと、先ほど重要インフラという単語が出たが、現在の重要インフラと将来の重要インフラは異なっていくのではないか。そういうことを含めて考えていければよいと思っている。

○　セキュリティ・バイ・デザインと言う言葉がはやっているが、現在今は家電製品のソフトウェアを作る段階でセキュリティを作り込んでいくと言うことをやっている。例えば、セキュアプログラミングのセミナを開催したり、セキュリティレビューできる人材を育成するなどである。家電製品の現場でセキュリティをやっている経験を生かして、本 WG に役に立っていきたいと思う。

○　セキュリティの監視を業務でやっている関係で、運用の現場での話を聞くことが多いが、一つ、運用という観点から意見を出せたらよいと思っている。ただし、現状では、このような運用の現場は色々な矛盾を抱えて疲弊していっているという現実がある。運用がだめになったときには、国のセキュリティを下支えしている部分が大きな危機に陥るのではないかと言う危機感を持っている。そういう観点から意見を出せたらよいと考えている。

○　IT サービスの立ち上げに 3 年程度係わってきたが、サービスを立ち上げるにあたっては、いかにお客さんに便利使っていただくかと言うことを考える。それらはセキュリティ・個人情報保護とは裏腹なところがあり、いつも法務とぎりぎりの部分で調整しながらやっているので、そういった観点から意見を出せたらよいと考えている。

○　アンチウイルスベンダのパターンファイルによるウイルス対策は破綻をきたしつつあるのではないかというのが色々なところで言われている。昨年のカスペルスキーからアンチウイルスベンダは敗北したのではないかというレポートが出ている。攻撃側と守る側の戦争のなれの果てには何があるかという事を考えると、自律進化してウイルスを発見し駆除できるようなテクノロジみたいなものがグランドチャレンジの中から生まれてくると夢があって良いのでは無いかと思っている。

また、国民目線の観点は重要だと考えている。家電・携帯電話・自動車等ユーザの広がりの中で、セキュリティに関するリテラシのあまり期待できない人がストレスを感じないセキュリティ対策が重要ではないかと感じている。

それらの二つの点から、夢のあるプロジェクトというのが出てきたらいいと思っており、そうしたものに貢献していきたいと考えている。

○　ワーキンググループの背景を説明した資料４について、ご質問はございますでしょうか。

特に疑問点はないと言うことなので、資料５－１、５－２、５－３について、あるいは自己紹介の中で強調した点などとも関連して、何かご意見のある方はいらっしゃいますでしょうか。

○　弊社で行われている家電製品のセキュリティを高める活動について言うと、家電製品が最近ネットワークにつながるようになってきて、セキュリティに対する要求が高まってきている。家電製品の多くをネットワークにつなぐようになってきているが、セキュリティの面では正直追いついていない部分も存在する。

私見だが、家電製品のセキュリティの状況としては、あまり被害が発生しておらず、エンドユーザにセキュリティの価値が認識されにくい。ただ、被害はゼロではないので、対策をしないで良いというわけではない。また、コスト・価格面の要求も厳しいものがある。

そうした中で、ライトウエイトのアプローチとして、できる限りのことを与えられた時間と予算の中でやろうと取り組んでいる。各セキュリティ対策の費用対効果を高める工夫をしており、セキュリティレビュー・教育を簡単にできて効果の高いものにして、費用対効果の大きいものから取り組んでいる。

そういったライトウエイトな取組でセキュリティの取組は大丈夫かという問いへの答えとして、セキュリティレビューを実施して、製品出荷前の 8 製品で 130 個前後の脆弱性をつぶすことができた。一つの脆弱性でも出荷後に修正すると最低 100 万円程度の費用がかかるので、レビューによって修正することのできた脆弱性の経済的効果はレビュー費用を上回っており、効果は上がっていると言える。

また、設計・ソースコードには脆弱性が入り込む余地があり、脆弱性混入に対する対策として、セキュリティを考慮した設計・セキュアプログラミングがある。他方、入ってしまった脆弱性に対してはレビューで対応している。

費用対効果の高いものとして、セキュアプログラミングのセミナを全プログラマに対して課しており、加えてレビューを行う専門家は 4 ヶ月の研修を受けている。以上が我々の取り組んでいるセキュリティ対策の概要である。

○　セキュリティレビューというのはソースコードレビューのことなのか。

○　ドキュメントレビューとソースコードレビューの両方である。但し、一部製品では片方しか実施しないこともある。

レビュー自体は一つの製品について半日～3 日程度なので、それを考えると、簡単なレビューでも十分に費用の面からは効果を発揮している。

○　セキュリティを考慮した設計について、教育コスト面についての費用対効果はどうなっているか。

○　例えば設計のレビューをやるときに、個人情報を暗号化するのだが、開発者がセキ

ュリティを学習せずに暗号化すると簡単に解ける暗号になってしまうが、それらをわかるようにするためにはそれなりの学習が必要になるため、カリキュラムの多くを暗号に裂いている。

コストという意味では、製品の特徴に合わせた教育を重点的に行うようにしている。

○　第三者提供のパッケージはどの様に扱っているかが知りたい。

○　バイナリで提供されているものについては、検査のしようがない。ソースコードで提供されているものについては、検査ツールにかけることもある。まず自社が係わる部分を重点的にやろうとしている。

○　HD レコーダで、実装の段階において設定を誤り、悪意の第三者によってスパム送信が可能となる、踏み台となりうる設定のものが、世界中にばらまかれたという事例がある。スペック・開発は間違えていなかったが、最後のパッケージの段階におけるミスにより被害が発生している。最近プラットフォームが多機能化しているので、見えていないパッケージについてどこでチェックするのか。

○　すぐには難しいが、セキュリティレビューができる人が増えるのが対策であると考える。まずはソフトウェア製造の部分でレビューするが、ある程度できるようになると、製品の出荷・保守まで含めたプロセスを見てセキュリティを保つようにしたい。そのためには、セキュリティがわかる人を育てる必要がある。

例えば、製品をパッケージに収めるまでに製造ラインでウイルスが入る可能性があるが、製造ラインの人の IT リテラシが低いとそうしたリスクも存在する。

○　セキュリティレビューに当たっては、脅威分析表を使うとのことだが、プロダクト毎に脅威が異なることは想定されるが、そういった点は意識しているのか。

○　製品にそれほどパターンが多いわけではないので、プロダクト毎に異なる脅威については意識することは少ない。

○　Web 系のアプリケーション等では、ある程度以上のリスクがあるシステムについては、ある複数のポイントにチェックを入れ、逆にリスクの低いシステムではもっと絞った箇所にチェックを入れるというのを考えているが、なかなか難しいので先ほどの質問をさせていただいた。

それとは別に、家電でも機能的には標準化できるところがあると思うが、共通化して危険を局在化するような取組はあるのか。

○　画像処理系の機能は共通化が進んでいるが、Web 系の機能はあまり進んでおらず、Web 系の脅威を局在化する取組は行われていない。ただし、共通的に使用されることの多いアプリケーションにおけるソースコードの巨大ライブラリについては診断ツールの貸し出しにより、ツールによる出力件数が減らす活動を進めている。

○　比較的セキュリティを担保するのに人で何とか使用とする方法が多いように聞こえるが、人力に頼らず機械化によって対応しようとしている箇所は有るのか。

○　現在、機械化しているのはソースコードのスキャンのみである。その他の機械化はやらないようにしている。理由は、セキュリティがわかる人が少ないので、人力によるチェック活動を通じてセキュリティがわかる人を増やすためである。例えば、レビューは一人ではなく複数でやることで知識の伝播が期待できる。

○　その活動はうらやましい。SI をやっていると、セキュリティを呼びかけてもそれどころではないという話になるところがあり、人手に頼ることに限界が感じられる部分がある。システムがどんどん複雑化するに伴い、制御できる範囲を超えていると感じられるところがある。何らかの自動化・機械化を行わないと体力のある会社は様々な手段をとれるかもしれないが、中小企業だと人手をかけるのはつらいという問題がある。もう少しセキュリティ分野でも機械の力を借りられないかという思いがある。情報セキュリティは未だに機械化されている部分が少なく、ドキュメントと人力で作業するという面があるのでどうにかならないかという思いがある。

○　緊急対応で出かけた際に、コードが脆弱でハッキングされたという事例があるが、仮にコードが脆弱でもデータベース側でアクセス制御ができていれば情報漏洩はしないし、ネットワーク側で防御するという手段もある。家電においては、そういったフェールセーフ的なアプローチはしているのか。

○　家電の場合はあまりそういうことはない。例えば、カーナビだとデータベースはあるが、単一のフラットな権限で動いているものもある。

○　ボットの話もあったが、一般の人はボットにやられるとカード情報やアカウントを盗られて大変だというのがあるが、企業に限るとボットにやられてなにかあるかと言うと、今現在は何もない。但し、go.jp ドメインにボットがあるとそれは温存しておいて、後にオークションにかけるなどストックしていることを考えられる。そうしたことから類推すると、家電でも個人のアカウントに入れるボットが売買対象になること

は十分に考えられる。彼ら犯罪者は何でも商売にする。

また、家電については、ネットワークにつながない方が良いものもあると考える。ユーザは何をするかわからない部分があり、例えば無防備な状態で、公衆が使えるインターネット接続ポイントに直結でつなげる事も考えられる。そういった倫理についてもこの場ではないかもしれないが、考えると良いのではないか。

○　家電製品を含めて、非常に高性能のコンピューティング能力を持った機器が計算機の姿をしなくなってきているという傾向がある。それを、ユーザが思わぬところにつないで回るというのは考えられる将来である。

○　先ほどの HD レコーダの話もそうだが、ユーザの思わぬ利用によって、いまだにセキュリティ上問題のある設定でインターネットに大量につながっていると言う事例もある。メーカがパッチを出しても、パッチを適用する人がいないという現実もある。

○　家電の世界では、コストが非常に重要になる。そのため、あらゆるケースを想定するのは不可能である。例えば、モバイルするものは色々なところにつないで使用される想定でレビューするが、据え置き PC 用のアプリケーションやテレビなどは家庭内での使用を前提にしている。中には、インターネットにつなぐとリスクがある旨の警告を表示して、OK ボタンを押さないとインターネットにつなげないものもあるが、全般的な基準は作りにくく、ケースバイケースの対応になる。

○　システムセキュリティとプロダクトセキュリティは分けて考える必要があると思っており、今までの議論は主にプロダクトセキュリティであり、売りきりモデルであるため、売った後のセキュリティについて非常にコストセンシティブというものがある。かかる費用が継続的なので、継続的なメンテナンスが必要となったときに、どういったお金の取り方があるかというのはビジネスとセットで考える必要があるのではないか。

また、セキュリティレビューの自動化については、弊社では短期的には出荷プロセスの前に必ず機械的なスキャンを通らないといけない仕組みを作るとか、プログラマとテスターとプログラムの使用を作る人がセットになってセキュアプログラミングを徹底する等の取組が存在する。品質管理を厳しくした経緯があるため、テスターの数が増えている。その数を長期的に減らすためには、抜本的な対応が必要である。

今手がけているのは、上流でのバグを減らすためにドキュメントを含まれているバグを減らせるかというのが重要で、スペックについても仕様記述言語で形式的に記述して、機械的にバグをチェックするのに取り組んでいる。そうすると、C 言語の限界があるため、長期的には OS そのものを抽象度の高い言語で作成し、スペックと実装の突

合もコンパイラで判別できるようにして、ソースコードに対して、システムベリフィケーションを行うOSも研究としては作成した。但し、既存OSとの互換性の問題があるので、将来的に製品化できるかは白紙である。

ただ、この点について、バーチャリゼーションの技術がここ数年で発達してきているので、特に安全性が求められるもののみを新 OS で動かして、従来のものは従来のOSで動かし、それらを安全なプロトコルでつなぐ事が可能になってきている。そういう意味では、研究を始めた6～7年前から考えると、コンパイラから新しく作ったOSの商品化については、楽観視できるようになってきている面はある。

○　自動化には二つ切り口がある。作る方を自動化できないかはソフト業界永遠の課題であるが、難しい。プログラミングの部分は抽象言語で書けば可能な面もあるが、上流工程とつなぐのがネックになる。他方、チェックする側はソースコードビットチェックなどがあるが、こちらも人が作ると言うことが限界になる。標準化して、そのパターン通りできているかをチェックする事になるが、それが限界になる。

機械に作らせれば安全になるかもしれないが、人が作っているため安全にならない。それをチェックの段階で機械的に行おうとすると、人が作っているため機械化しきれない。と言う矛盾したところがあり、それが一つの課題になるのではないか。

コストの面の話について言うと、製品ではコストの面で難しいという話があったが、ビジネスアプリでも同様で、セキュリティをしっかりやるから費用が増えるというのはまだ受け入れられがたい現状にある。最近は顧客のセキュリティ意識も上がってきたので、しっかりと説明できれば費用転嫁できるが、どう説明するかというやり方はいまだに確立されていないところもある。ここは開発側の課題である。

ボットについても述べさせてもらうと、初期のボットにやられた会社を知っているが、二週間で300台のPCを再インストールした。3ヶ月ぐらいは何もしなかったのだが、その後に当時の最新ウイルスをダウンロードし始めたため、発見された。ボットを足場にして色々なことが可能になってしまうため、ボットが入ること自体が企業にとって大きなリスクである。その後、不審な通信を監視する体制をとるようになった。通信だけ見ていても難しいが、どれだけ多くの切り口から見られるか、可視化できるかというのがポイントだと考えている。ただ、あまり見せすぎると情報が多過ぎて整理がつかないという問題もあるので、どの様に自動化して必要な情報を見つけられるようにするかというのが課題であると感じている。

○　OS をモデルチェックできるように機械的に生成すると互換性がなくなるというのはどういう事を想定しているのか。

○　研究で作成したOSはWin32のAPIセットを持っていないため、既存のアプリケー

ションは動かない。OS そのものも C#のマネージドコードで書いているため、その上で動くアプリケーションも、メモリをアプリから自分でいじらないような書き方しか許さないような作りにしている。そのため、Windows 上の.Net のアプリをコーティングすることは比較的簡単できるが、C で記述して自分でメモリマネジメントをするようなアプリケーションをコーティングすることはできないようなものになっている。

○　それを前提に、VM にするとそれを商品化しやすいというのはどういう事を想定しているか。

○　現行 OS は現行 OS で動かして、新 OS は新 OS で VM 上の別のパーティションで動かして、現行 OS と新 OS との間でトラステッドチャネルを使用して、コミュニケーションをするという事を想定している。

○　別の観点で質問だが、C#で作っていくときに、OS は作れるかもしれないが、分散システムとして組んだときに別の要素が入るのではないか。プロトコルを一体としてモデルチェッキングできないという問題があるだろうが、そのあたりはどう考えているのか。

○　その OS はマイクロカーネル風に作っていて、型と通信の状態遷移を仕様記述言語で書けるようにしている。通信の状態遷移に対してモデルベリフィケーションをかけるので、ソースレベルで、同じ OS インスタンスの中であっても、他のコンポーネントとのやりとりであっても、同じ仕様記述言語で書けば、そこに対してモデルベリフィケーションかけることは可能である。

この世界は昔見たものがたくさんあって、ポイントはいかにそれを使い勝手を良くするかというのが大事だと考える。

○　そのほかに、昔動かなかったものが PC のパワーアップに伴って動くようになったと言うこともある。必ずしも個々のアイデアはそこまで新しくないというものも多い。

○　今の議論を聞いていて思ったことだが、セキュリティというのは各論の積み上げになりがちで、言うならば分子はあるが分母が不明確な状態になっている。現状でたくさんのセキュリティ対策があるが、ローカルではともかく正規の一覧は見たことがない。本当にそのままで良いかというのは感じている。

顧客によく言われることは、セキュリティはどこまでやればいいかがわかりにくい。やらなければいけないことはわかるが、どれからやればいいか、どれをあきらめねばならないかがわからない。

国の基準でも分母に該当する部分がないので、運用現場ではセキュリティは守らねばならないが、コストはかけるなと言う要求に晒されている。分母がはっきりしていれば、ここまではやるというように予算化しやすいのだが、それもできない。国のセキュリティ基準も複数有って、粒度等もバラバラで統一できていない。

今までのように個々の技術について検討していくのも重要だが、技術のまとめについても考える時期に来ているのではないかと思う。

○　目安がないことで不安になるのは合意するが、分母が無いことに対する対策について画一的に決めた方がよいかは、非常に慎重にあるべきだと思う。結局脅威をどこまで考えるかだが、これまでの歴史を見ると一番安全な技術が普及したわけではなく、トレードオフを考えたものが普及している。最終的には市場での調整を経なければならないので、国としては分母を確定する役割があるのか、他の方法によるセキュリティ向上のファシリテーションがあるのかは別途検討する必要がある。

○　分母をまとめたものを作るのが難しいのは理解できるが、分母を貯めて、整理していく仕組み・考え方は整理する必要はあるのではないか。新しい手口が出てくるため、それを整理していく考え方は確立すべきであると考える。

○　これまではセキュリティ対策はパッチワーク的に行われていたが、今何がおきているかというと、今までやってきたセキュリティ対策を見直すとバラバラになっている。例えば、ファイアウォールを見ている人と、IDS を見ている人と、アンチウイルスを見ている人が別々だったりする。よく考えると、今の脅威はそれらを一体として守らねばならないのに、運用がバラバラで統一する仕組みがまだできていないというのが見受けられる。もしかすると、新しい脅威でも既存の対策をうまく運用できれば防げるものがあるかもしれないので、今までやってきたものを再整理するというのは有効だと考える。

○　利用者の人まで含めて話が通じてほしいと思う。そういう意味ではどこまでやればいいのかというのがないと、特にセキュリティにおいては難しいものがあるかもしれない。

○　グランドチャレンジのテーマについて議論していると、それは何を作るのかと思うことが多いと思われる。例えば、今の家庭ネットワークは自分でマネージしにくくなっており、体感性がないというセキュリティの特徴もあって、自家ネットワークがどれまで安全かが実感できなくなっている。箱を一つネットワークに接続すると、光る色でどの程度安全かわかるような夢のような箱を作るにはどうすればいいか、そうい

うコスト・マーケット・機能の面からの手軽さを持ったものはどうすればターゲットになるのかというテーマをずっと考えていた。

他方、先ほどモデル論の話が出ていたが、背後にはオートマティックコードシンセサイジングの技術がある。これは成長するシステムを仮定しているが、今までのセキュリティにおいては成長するモデルは考えられていない。健全にセキュアシステムを成長させる環境・健全な成長を監視するシステムは大きなチャレンジだと考えている。5 年後にどうなるかを一言で表すテーマを言えるとこの WG の先行きも明るいと思う。セキュリティの会合では不満が良く出てくるが、そこからテーマや、政策や、予算の基になるものに持って行くための化学変化が重要だと考えている。

○　生物学で出てくる免疫論のように、放置しておいても成長していくメカニズム・成長を監視するメカニズムが有れば、そういうものが有れば、作る側もエネルギーを投じなくても良くなるという人間と機械のバランスが達成されるかもしれない。また、リテラシをあまり持たなくても良いユーザがいるというようにバランス良い状態が達成されるので、そういった方向もグランドチャレンジで検討していただきたい。

○　一つ一つの状態を追いかけるより、全体としてみると大丈夫か否かはプログラムより、その周辺、例えばデータベースなどに出てくるような気がする。

○　センサーネットワークの研究をやっているが、センサーネットワークというのはセンサーの数を増やせばより多くの状態をとれるようになるが、プロセスは大変になる。これと同じように、セキュリティでもモニタリングをきつくやると全部見えるかと言うとそうとも言い切れないようなものがある。多くの対策を導入してもゴミの山ばかり増えて、実態像がわからないという事態が発生しうる。それらをどうやって再構成して役立てると言うノウハウはあるが、そういったノウハウに基づいてもある一面についてはわかるが、その他の面についてはわからないと言うことが起こりうる。

センサーネットワークと同じ問題が起こっていて、ちょうど部屋に温感センサーを 100 個つけても、人間がいることはわかるが何人いるかはわからないようなものである。別の知見・メカニズムによらないと、実態がわからない。そういった成果の再発見をまじめにやらないといけないというのは常に思っている。先ほどいったネットワークに差し込むとネットワークの危険度がわかる箱という様なところに集約できるように感じられる。

○　クラウドコンピューティングは結構おもしろいと思っていて、今弊社では一人で 5,000 台のサーバをメンテナンスできるようなミドルウェアにしているが、某社のネットワーク監視に弊社のロボット用のミドルウェアが使用されている。SMMP のイベント

をロボットへのセンサー入力に見立てて動くような仕組みがある。

クラウドコンピューティングのために、マイクロソフトやGoogleがコンテナ単位で購入しているPCと、家庭用の1 TBのハードディスクを備えたテレビを比較するとほとんど同じようなものである。サーバ側のクラウドでは1台1台を別々に扱うのではなく、集合として扱うことで一人が数千台のサーバを管理できるようになっているのと同じく、情報家電においても上のレイヤで新技術が出てきて、何が起こっているかを把握できる技術が出てくる可能性は十分に存在する。

○　それと関連して、家電の場合は売りきりで良いのかという話は産業全体に係わる可能性がある。

○　私の場合についてだが、デジカメを使うようになってから、紙の写真に比べて昔の写真がどこにあるかわかりにくくなってきている。データの信頼性というのが過渡期に来ているように見受けられる。消費者からわかりやすくデータを補完できる仕組みも考える必要があるのではないか。

○　クラウドコンピューティングの話や免疫の話が出ていたが、人間の細胞はどうしようもなく傷つくと自分で死ぬ仕組みがある。今は一つのコンピュータで一つのことをやっているが、複数のコンピュータで複数の仕事をやることで、どこかにセキュリティ上の問題が発生してもそこを落として助かるというのが有るのではないかと考えている。プラットフォームの冗長性を格段に挙げるというのがテーマとしてあるのではないか。

○　今までのやりとりとしては、テーマ的なところが多かったが、研究管理も問題になっているので、そこについて触れておくと、個人的には、大きなお金を投資する際に技術だけを見ていて、ビジネスにつながるかと言うことを見てこなかったという反省がある。グランドチャレンジと言うことでスケールの大きいことをやるならば、できあがったものが世の中に還元できる仕組みを考える仕組みが重要だと思う。

宇宙船チャレンジャーの爆発では技術者が部品の不備情報をエスカレーションしようとしたときに、情報が落ちてしまったという組織論にまつわる話がある。それらと同様に、グランドチャレンジがサブテーマに分かれたときに、それらをつなぐ組織的な情報共有の仕組みが必要だと考える。

○　最後のものはプロジェクトの進め方における話ですか。それともアウトプットとして出てきたものに係わる話ですか。

○　前者の方である。

○　メインフレームに係わったときに、動いている人間が１００％動かなくとも、ものが１００％できるようにするためには、プロジェクトマネジメントが大切だと感じられたことがある。人間システムを含めるとどこかがダメでもどこかが動いていたのかもしれない。

プロジェクトの進め方について言うと、日本のプロジェクトは計画性が強すぎると言われている。アジアの国は日本をよく見ており、計画性が強いことも知っているので、プロジェクトの流れを変えたとき、日本ではできないでしょうと言われて悔しい思いをすることがある。韓国も前大統領の頃は、８０％は予定通り、２０％はフレキシブルにやって良いと言うことがあった。韓国については、大統領が替わったため、制度も大きく変わるとは思うが。

○　身近な話だが、企業では PMO を設置することが結構あると思う。当事者間では危機的状況があっても見えなくなることがあり、利害関係の薄い PMO がフェーズ毎に方向を修正すると言うことが定番的に行われていると言うこともある。
別目線で、軌道修正をする権限を持つ人をおいておけば終わってみて成果が全然挙がっていないという事態を防げるのではないか。

○　システム開発の場合は PMO が機能することがわかっていて、政府でも GPMO と言うのがある。他方、技術開発、特に政府のお金で行われる技術開発は開発経費より研究経費という色合いが濃い。プログラムマネージャ制度等色々あるが、誤解を恐れずに言うと、政府の予算を取りに来るのは一流のチームが多く、要素技術によってテーマが考えられていることもあるので、評価をするのは他の領域の一流か、同じ領域の二流以下の人になってしまう。そうすると、プログラムオフィサーが自信を持てなくなってしまい、当初計画通りにやることが重視され、リスクテイキングがなされず、適切な助言が行われない。

逆に自由度が上がったときに何が起こるかというと、仮に勇気を持った PMO がいても、会計検査院対策が大変になる。予算根拠・支出適正性についての事務作業が莫大になる。そういう状況をいかにうまくやるかというのが、システム開発と研究開発の差ではないか。

○　文科省の科研費で研究をやっているが、手練れがいてきちんと評価して、やっていることをおかしいと言いにくい雰囲気はある。今の研究において、システム的なことをやっている人の関心はコンピューティングリソースのある部分をモニタリングに使って、システム全体が自立的に動くことについてについて存在している。

モニタリングしたり、トラブルをローカライズしてどう対処するかと言うことをやっていく必要があるのではないか

○　研究開発のプロジェクト管理については、2 点質問がある。一つは、国と同じ問題が民間では起こらないのか、起こるとしたらどう解決されているのか。もう一つは原則として、計画が変更できるようなルールにした場合、なぜ事前に見通せなかったのかと責められる。その時に、新たなルールにおける、合理的な説明ができるのかと言うことである。

○　我が社のごく一部の事例になるが、研究と開発は分かれている。開発というのは予定通りにものを出すと言うことに集中している。全く違う脅威が出てくる、全く製品の環境が変わると言うことはあるが、対応は大きく分けて 3 つある。一番簡単なのは地道な継続的改善であり、もう一つは長期視野の R&D では学術コミッティーの中で評価するものである。そして、それらの研究と市場ニーズの間を埋めるのは企業買収である。民間企業の場合は、予測でいない変化に対応する適応は企業買収というのが多いが、これはそのまま、政府の研究開発に適用できないので、考える必要がある。

また、成果利用のプロセスの独立と言ったときに、成果をどう位置づけるかが重要になる。コードベースの再利用なのか、コンセプトの再利用なのか、人材のスピンオフなのかで全く異なっている。コードベースの再利用を無理に期待すると、表向き 10 年と言っていても発想のスコープは短期に向かざるを得ない。

ソフトウェアについてはデプロイメント、サステインエンジニアリングのところが、抜けがちになる。お金がかかるのはその段階なので、そこはビジネスなので国のやることではないとなりがちである。

成果利用をなんなのかと言うことをきっちり決めていくと、そこから逆算してどこで方向転換するというのも見えてくるのではないか。

○　ソフト開発で PMO がきちんと機能できるかは大変なところもある。ソフト開発も順調にいっていると良いが、何か問題が起こったときにどう PMO がきちんとアドバイスしてプロジェクトを修正できるかというと、うまくいくケースもあるが、問題の指摘にとどまって、うまく修正できないこともある。おそらく、開発でも問題が起こったときには抜本的にひっくり返す発想が必要である。

大きな問題があったときに、それを修正するのは、オーバーホール的な発想ではなく全部ひっくり返して考える必要があり、それができるかどうかが PMO として機能するかどうかである。その部分がうまくいっていないで、当初の計画にしがみつくケースも発生している。いわば、民間でも PMO が会計検査院化しているケースがある。
日経 IT の Web サイトに JISA 会長の浜口氏のコラムでも、日本のシステム開発の問題

として、ウォーターフォールモデルにしがみつくことの問題点が指摘されており、すごく納得がいった。今の開発モデルではチャレンジを行うことが難しく、保守的にならざるを得ないのではないか。これは研究開発にも通じるが、変えなくてはならないと思う。

○　民間だからと言って、簡単にいくと言うことはないことがわかった。プログラムオフィサー本当にスイッチできるかどうかは難しい。スイッチしたら最終評価でスイッチしたことがあまり評価されないという事例も存在する。ただ、そういうことが問題だという意識は出てきているのでは無いかと思う。諸外国でも全部がうまいこといくわけではない。

○　プロジェクトの途中で状況が変わって変化すると言う話において、どう変化するかと、その変化の承認プロセスは別の話であると考える。承認の話は研究テーマとは別だが、研究を進めるには仕組みも変えないといけないので、難しい話だと思う。

○　大きく二つの問題があり、専門家で問題意識を共有していれば、コミュニケーションコストが下げられて評価者と研究者で手を携えて方向転換ができるはずという人に起因するコミュニケーションの問題がある。

他方で、セキュリティはすでにあるシステムに対する脅威として出てくることを考えると、全く新しい技術を世に問うてそれで安全が確保されることは非常に少ない。そうすると今動いている脆弱性を抱えたシステムを同方向転換するかと言うことが問題になる。多くの評価されたセキュリティ技術は後から入り込んでいくところのボトルネックを解消したものであるというのは言えると思う。

○　情報システム自身の本質的な問題、2000 年問題の時のように今動いているシステムが問題になるということですね。

○　サービスをやっていた人間の観点から感じたことは、サービスというのはそこまで目新しいサービスはあり得ず、実際の生活の中でどう使っていくかというのが問題になる。

セキュリティにおける究極の目標は、大きな変化が少ないことが多いというのは、サービスの世界でも同じで、生活の中で使われるようにいかにするかという観点で見るべきだというのが実際に係わった感想である。

技術をどう生活の中で使うというのが重要であるが、今までの議論で抜けていると感じたのはコスト的な視点である。セキュリティ、サービス、コストの三つの視点で考えるともっとおもしろくなると思う。

コストの視点を除くと何でもやればよいとなりがちだが、何かを捨てて何かをとると言うことをやる必要があると考えている。

サービス的な視点で言うと、そこまで新しいものが出て来るとは思えない。そんなに難しいものではなく、今あるものが変化することが多い。そういう視点で考えるとセキュリティに新たな視点が入ると思う。

○　ライトウエイトというコストの話もあったが、そういうのも考えると良いと言うことであろう

○　研究において、成果目標の組み替えは必要になるが、そればかりをやっていると今度は何をやっているのかがわからなくなってしまう。

今やっているのはメインストリームの計画とは別に潜水艦と呼ばれる小さいものを作っておいて、プロトタイプ・論文ができたときに発表してメインストリームに持ってくると言うことである。

グランドチャレンジにおいては巨大潜水艦を作るという感覚が近いのではないか。また、教育でプロダクトセキュリティの底上げを図るのは非常に納得できるが、セキュアプログラミングとして挙げられた例は C と C++である。言語を統一するとプログラミングの使い方でセキュリティを向上させることは可能だが、言語が違ったときに言語依存のものをどう解決するかという問題が発生する。

○　補足すると、我が社の場合は言語依存に関係ない部分のセキュリティを優先しているという現状がある。

以上