CYBER GRID JOURNAL

サイバー・グリッド・ジャーナル

VOL.6

特集

ミニチュア技術、IoT

未来の社会を創造する新技術へのラックの挑戦

CYBER GRID JOURNAL VOL.6

TABLE OF CONTENTS

巻頭言

先進的な IoTセーフティ・セキュリティの 研究開発に挑む

渥美 清隆
サイバー・グリッド・ジャパン
IoT技術研究所長

2014年あたりからIoTという言葉が広がり始め、2017年にはすっかりIoT元年という雰囲気になりました。IoT（Internet of Things）に「モノのインターネット」などというおかしな訳語を付けていたときもありましたが、これは最近あまり見なくなりました。2016年以前に考えられていたIoTと呼ばれるデバイスもほとんどはルーターかウェブカメラで、「冷蔵庫がネットワークにつながった」といって話題になるほど、牧歌的な時代でもありました。

ウェブカメラも含めて2015年頃からネットワークに接続される機器が急激に増加しています。総務省の統計によれば❶、2013年には約112億台の装置がインターネットに接続されていたものが、2021年には約348億台に増加することが予想されています。また、接続機器の種類も豊富になり、スマートスピーカーのようなコンシューマー製品や、医療、自動車、産業ロボット、航空、軍事に至る相当広い範囲にまで接続が広がっていることが分かります。

一方、この1、2年でIoT機器に関係するセキュリティ事件・事故も急速に増え始めました。Miraiやその亜種は、IoT特有のマルウエアというよりも、IoT機器の特徴に合わせて作られたパソコン用のマルウエアと同等のマルウエアということができます。パソコンと同等のマルウエアならば、パソコンと同等の対策をすればよいのではないか？　という声も聞こえてきそうですが、残念ながらIoT機器のコンピューターは動きがとても遅く、記憶装置も非常に小さいため、アンチウイルスソフトを載せることはできません。また、問題が発生したときに、パソコンならディスプレイに問題を表示することが可能ですが、IoT機器の場合はそれを表示する場所がありません。IoT機器がマルウエアに感染したとしても、ユーザーが知る機会がないのです。結果としてIoTマルウエアは大規模なネットワーク障害を引き起こし、社会問題になるまでに至りました。

IoT機器のマルウエアは今後、ネットワーク障害を引き起こすだけでなく、電力網を停止させたり、ネットワークに接続された自動運転自動車を乗っ取って乗客に怪我をさせたり、IoT化された工場を爆発させたりといったように、人々を傷つけ、最悪の場合は死に至らしめるようなものが出現する可能性が高いです。私たちIoT技術研究所としては、IoTマルウエアからIoTをどう守るか、また不正なIoTから、どのようにすれば人命が守れるのか、社会秩序が維持できるのかを考えながら、今後の研究開発活動に邁進していきたいと考えています。

❶ 総務省平成29年版情報通信白書：爆発的に増加するIoTデバイス
http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h29/html/nc133100.html

特集

ミエナイ技術、IoT

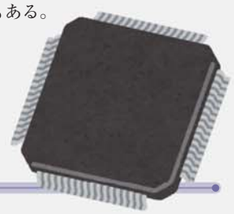
未来の社会を創造する新技術へのラックの挑戦

IoTエッジデバイスエンジニアの苦悩の日々

サイバー・グリッド・ジャパン　IoT技術研究所　リサーチャー　木田 良一

IoTと呼ばれる世界は、広大なテクノロジーによって構成される。エッジデバイス[1]側から見ると、リッチなリソースを持つクラウドと比較してエッジデバイス側は恐ろしく貧弱な環境下で製品機能を満たす必要がある。

IoTエッジデバイスのハードウエアを設計するエンジニアには、常に原価を意識して設計をすることが求められる。設計したプリント基板上に余分な回路を載せることは許されず、ましてやプリント基板上に手作業で配線をすることは許されない。手作業で配線をするために工場の従業員の手を使うことがすなわち価格（コスト）につながってしまうからだ。また、保護回路を設けたくても保護回路のために必要とされる部品価格を会議の場で徹底的に叩かれ、仕方なく製品の価格を抑えるために保護回路を外すこともある。



IoTエッジデバイスのハードウエアエンジニアの最大の敵は、そのデバイスのソフトウエアを設計するソフトウエアエンジニアであったりする。彼らのスキルセットを最大限に生かし、製品の販売を遅らせることなく設計を完了させられるよう、最大限の努力を払わねばならない。ソフトウエアエンジニアが、「プログラムが動かない理由が不明だ」と言えば、昼夜を問わずハードウエアに問題がないかを検証しなければならない。メモリが足りないと言われれば、そのメモリを搭載可能なようにプリント基板を設計し直さなければならない。特定のオペレーティングシステムを動かしたいと言われたときには、それが動作可能なハードウエアを設計しなければならない。例えば世の中のソフトウエアエンジニアが魔法使いであるならば、ハードウエアエンジニアは神と呼んでもよい。それだけ万能でなければ、IoTエッジデバイスの設計は不可能なのだ。

ハードウエアエンジニアの苦悩と格闘は今に始まったことではない。その苦労はるか昔、今の企業の経営者たちが現役のハードウエアエンジニアであった時代から何一つ変わっていない。変わったのは、テクノロジーの進化とともに複雑化するマイクロプロセッサーとその周辺機能を構成するハードウエア、そして、年々分厚くなるハードウエアの仕様書である。その昔、200ページに満たなかったハードウエアの仕様書は、テクノロジーの進化とともに肥大化し、今や2000ページにまでなっている。その電話帳のような仕様書を読み解いて製品の機能を満足するようにプリント基板を完成させるハードウエアエンジニアは、まさにIoTテクノロジーの中において神ともいえる存在だといっても過言ではないのだ。

IoTエッジデバイスのソフトウエアエンジニアは、IoTという単語が生まれる前から苦難の道を強いられている。彼らは、製品の原価を下げるために無理やり設計されたハードウエア上で、製品の機能を満たすようにソフトウエアを設計する義務をなすりつけられている犠牲者でもある。ましてや、テクノロジーが進むとともに制御手順が複雑になったハードウエアの制御を一手に任され、納期という重責に耐えながら日々、実務に追われている。

彼らの不幸は、たとえ便利な仕組みがあったとしても、必ずしもそれを使えるわけではないという点にある。ハードウエア上に組み込むオペレーティングシステムによっては、ハードウエアの互換がないために効率的な暗号アルゴリズムの実装を断念し、ソフトウエアのみでシステムを実装することもある。また、知的財産という壁に作業を阻まれてしまうこともしばしば起こりうる。使えるメモリのサイズに制約がある場合には、オペレーティングシステムそのものが限定される場合もある。そのようなときのために数百キロバイトという小さなメモリで動作させることが可能なTRONと呼ばれるオペレーティングシステムを使うも、多くのソフトウエアエンジニアが経験しているPOSIXと呼ばれる機能を持ったオペレーティングシステムとは語法すら異なるインターフェースを読み解き、IoTエッジデバイスの機能を満足させなければならない。組み込みソフトウエアのエンジニアが扱うソースコードの量は年々増えている。それゆえ、ソフトウエアエンジニアが見渡すソースコードは広大なものである。先ほど、ハードウエアエンジニアが扱うハードウエアの仕様書は2000ページに及ぶと述べたが、ソフトウエアエンジニアが見渡すソースコードはときとして100万行にも及ぶことがある。果たして、この広大なソースコードを組み合わせたときに製品として問題は発生しないだろうか。そして、いつこの実装が完了するのか。ソフトウエアエンジニアは常にこの精神的葛藤と戦い続けている。そう、彼らは戦士なのである。

今、IoTテクノロジーにおけるこうした戦いの中、クラウドからエッジデバイスにわたる広範囲のエンジニアに共通の敵が現れている。IoTを悪用しようとする攻撃者に対するセキュリティである。IoTエッジデバイスはもともと、狭小なメモリでも動作するように設計されている。IoTという言葉が生まれる前のデバイスに至っては、パッチを適用するということは考えられていないものもある。その当時のデバイスでは修理・交換という手段が用いられることが多かった。なおかつ、企画・設計の段階においてはデバイスの安全性というものに気の遠くなるような配慮をしているが、それが悪用されるだろうことは積極的には考慮されていなかった。IoTエッジデバイスというインターネットにつながる機器となった時点で、インターネット経由で不正利用を働く人間の存在が前提となる。そういったことに対して十分な配慮をする必要性に迫られてしまった。そういう時代になったのだ。

[1] インターネットにつながる監視カメラやネットワークレコーダーなどのことをいう。

セキュリティで問題が発生することは、経営者にすれば死活問題である。自社のブランドに影響を及ぼすものであり、結果、経営者の責任が進退問題にまでつながってしまうこともある。そこまで問題が大きくなると、会社が存続できなくなってしまうかもしれない。

そうなった場合、従業員も蚊帳の外で眺めているだけでは済まない状態になってしまう。景気は緩やかな回復基調にあるといわれているようだが、やはり現実は厳しい。そう、セキュリティ対策は結果として家族を守るためにも必要となっている。

今、セキュリティに配慮したIoTエッジデバイスの開発は必須となっている。そうなっていない機器はいずれ淘汰され、ゾンビのように世の中の片隅で生き恥をさらし続けるだろう。IoTエッジデバイスエンジニアは、今までの苦労の上に、セキュリティをそれぞれ考慮しなければならなくなる。セキュリティだけを別に考えることはできない。なぜならラーメンのトッピングを全て

無料にするのと同じことを自分たちの設計したエッジデバイスに課すことになるからだ。無原則に割り引いてしまっては原価が合わず、自分たちの給料すらままならなくなってしまう。そういう意味で、セキュリティというものを従来の苦労の中に混合配分して製品を完成させなければならない。

ここまで読み進んでいただいた皆さんには理解ができると思うが、IoTエッジデバイスのハードウエアエンジニアならびにソフトウエアエンジニアは多大な汗と涙と愚痴の末、さらには気の遠くなる心身の葛藤を伴いつつ、製品を仕上げている。そこに、製品を企画した企業や製品に携わった設計者の意図にそぐわない使われ方が利用者によってなされる可能性に配慮する責任も加わるのである。セキュリティを踏まえたということが製品のカタログに明記されないとしても、利用者がセキュリティ面で安心できる製品を当たり前に買えるようにしなければならない。これが結果として利用者を守るのである。そんな使命をIoTエッジデバイスエンジニアが背負うことになる。

ラグビーワールドカップや東京オリンピック・パラリンピックを控え、日本が世界からより注目されることを考えると、日本にあるデバイスが悪用のために攻撃されないとは限らない。平昌オリンピックでシステム不具合が発生したときにドローンが飛べなかった教訓は、活かされるべきである。一方では、カタログに載らないセキュリティ機能を今までの設計過程に忍び込ませる形で実現させるべく努力をしなければならない。そう、IoTエッジデバイスエンジニアは意識せずとも実は日本を守っているのである。

セキュリティ対策というと、また新しく大掛かりな投資が必要なのではないかという疑問が生じるであろう。しかし、もう一度見直してみていただきたい。ハードウエアエンジニアも、ソフトウエアエンジニアも、あらゆるIoTエッジデバイスのエンジニアはIoTという冠が付く前から、自分の設計した製品に何らかの問題があったときにはまず利用者の安全を優先するように設計をしている。つまり、得体の知れないことが起きる前に製品を安全な方向に制御する努力を最優先しているのである。エンジニア全員がセキュリティのプロフェッショナルではないにもかかわらず、安全を考慮した設計というものがすでに日本の開発現場には染み付いているのである。ニュースでIoT機器が悪用されるケースを見るたびに、IoTエンジニアは皆、胃が痛くなる思いを必死にこらえている。

IoTエッジデバイスのエンジニアは日々、まるで仏教徒が阿弥陀様に救ってもらえるまで修行を続けるが如く、新しい技術とセキュリティの鍛錬を続けている。ただし、仏教徒は阿弥陀様に救ってもらえるが、エンジニアを救ってくれる者はいない。せめてもの救いは製品を買ってくれた人からの温かい言葉だ。それが、彼らエンジニアのモチベーションとなるであろう。

IoT機器の安全性はどうやって破られるのか

サイバー・グリッド・ジャパン　IoT技術研究所　リサーチャー　木田 良一

◆ IoT機器への止まらない攻撃

2018年4月、全国の自治体に設置された監視カメラが何者かにハッキングされ、監視カメラのパスワードや画像に変更が加えられるというセキュリティ事故が発生しました。過去数年を見ても、監視カメラに代表されるIoT機器のハッキングによる情報の改竄被害は相次いでいます。特に、Miraiと呼ばれるボットによってIoT機器が大量にウイルス感染する攻撃があった2016年以来、巧妙に手口を変えながらIoT機器が攻撃される事例は後を絶ちません。

◆ 身近に存在するIoT機器攻撃の手口

では、セキュリティ事故はどうやって発生するのでしょうか。市販のルーターを例に、攻撃者（ハッカー）の視点による乗っ取りの様子を紙面上で簡単に再現したいと思います。

ルーターの筐体には、初期設定で使用するユーザー名とパスワードが貼り付けられています。図1参照 これを設定画面で変更しない限り、同じユーザー名とパスワードでルーターの設定を書き換えることができてしまいます。多くのIoT機器の乗っ取りはこういった情報をもとに行われます。ユーザー名とパスワードは必ず変更するべきです。

図1　ユーザー名とパスワードが記載された製品筐体

◆ データベース化されているIoT機器攻撃の手法

ユーザー名とパスワードを変更した場合でも、ルーターのファームウエアが最新でない場合には古いファームウエアのバグ（脆弱性）を狙ってルーターの乗っ取りが試されます。

多くのサイトでは、ルーターの品名をもとに、使用されている制御プロセッサーの型番やルーターのファームウエアのベースとなるファームウエアのソースコードを調べることができます。同様に、ルーターに使用されているソフトウエアのバージョン別に、バグの情報も調べられます。こういった情報をもとにコツコツと攻撃を続けることで、ルーターを乗っ取ることが可能となってしまいます。

◆ バグをなくしても攻撃できてしまうのはなぜ

さて、ここまでに記載した点の全てに配慮しても、乗っ取られてしまうケースもあります。想定される手口として、カメラに設定したパスワードを解析された場合が挙げられます。では、どうやって解析するのでしょう。自治体単位で個別のパスワードを利用せずに全て同じパスワードを設定したと考えてみます。そうして、こっそりと一台の機器を影響のない範囲で入れ替えて持ち出した場合。しかし、どうすればその機器から情報を抜き出せるのでしょうか。

◆ 外見からは想像できないIoT機器

機器を実際に分解してみることにします。分解といっても破壊するのではなく、元通りに動く状態にして、その設定のまま機材を元の場所に戻すことを前提にします。

機器の筐体は、一般家庭にある道具でも外すことができるネジで組み上げられています。図2参照 たいてい、ネジはユーザーの手に触れられないよう、テープなどで隠してあります。丁寧にテープを外すと、ネジが現れます。このネジは一般的なプラス（マイナス）ドライバーを使って外すネジと異なる、六角形のネジです。これを全て取り除きます。

図2　ネジ穴の様子

そうすると、筐体を大きく2つに、さらにパネル面を合わせると機器を3つに分離することができます。配線なども丁寧に外すと、ルーターの制御の中枢となるプリント基板が現れます。図3参照

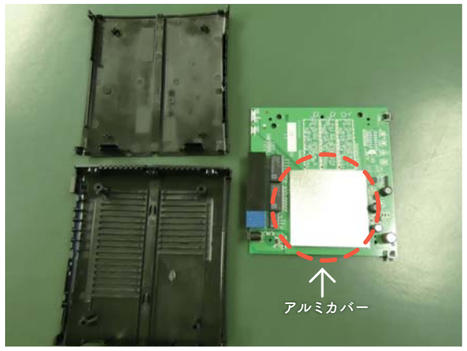


図3 基板と筐体とに分離された製品

プリント基板はアルミでカバーされています。流通する際に他の機器（特にテレビやラジオなどの一般家電）に電波などの影響が加わらないよう配慮されていると考えられます。これはマイナスドライバーやピンセットで簡単に取り外せます。中にあったのは、専用の制御チップとメモリです。図4参照

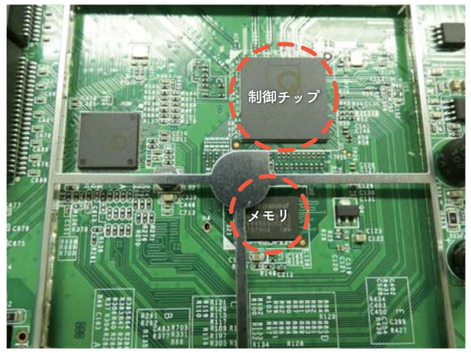


図4 制御チップとメモリ

この状態からメモリの内容を読み出すことは不可能ではありませんが、元に戻せるように扱うには高機能なハンダ処理を行う機材が必要となります。今回は、ここはそのままにしておきます。他に手段があるはずです。こういった基板を見るときには経験則による「このピンのコネクタがここにあったらこういった信号が存在するはず」といった勘が頼りになります。おかしな話ですが、ハッカーにとっては勘も重要な技術となります。

基板の右上を見ると、なにやら意味不明の14ピンにも見える2列×7ピンのような痕跡があります。図5参照 これは何でしょうか。これは、JTAGと呼ばれる共通規格による信号線を取り扱うピンです。調べると、何本かは共通してGNDと呼ばれる0Vの信号となっているようです。テスターで試してみます。

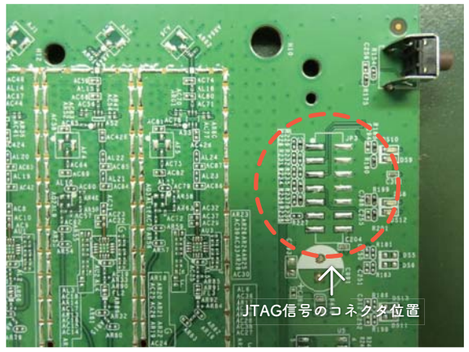


図5 JTAG信号のコネクタ位置

確かにGNDのようです。どうやら、チップの情報をもとに細かなハードウエアの情報を用意すれば、このピンから内部に設定されたユーザー名、パスワードを抜き出すことは難しくないようです。ですが、JTAGというインターフェースを利用して、このような機器に接続可能なデバッガーは、このルーターのような機器が対象となる場合、数十万円から数百万円といった高額なものとなります。そこまでの機材を駆使してルーターを解析しようとするのは、もうどこかの企業にいる産業スパイのような存在かもしれません。

もっと簡単な方法で、機器のメモリから設定されたユーザー名とパスワードを探し出す方法はないでしょうか。基板をもう一度見ていきましょう。先ほどの14ピンの端子の下に4ピンの端子があるのが分かります 図6参照 。ユーザーが使用しないのに、しかも内部的にもそこから配線でどこにもつながっていないのに、ピンが存在しています。このピンは何でしょうか。

図6 シリアル信号らしきピンヘッダ

端子の接続をテスターなどで調べてみると、どうやらこのピンはシリアルポートのようです。送信信号、受信信号、電源（+5V, GND）で構成されているようです。このピンにケーブルを接続して、Windows上で端末ソフトウエアを起動してみましょう。図7参照

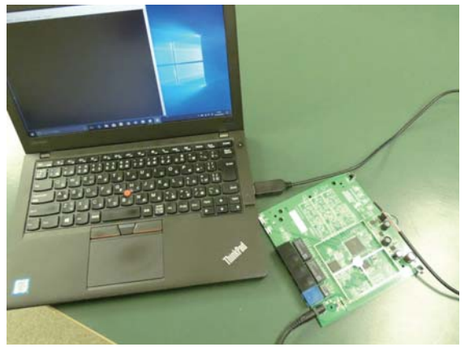


図7 シリアル通信をするようにケーブルを接続してみた

すると、なにやらコマンドやその応答といった文字列が表示されました。図8参照 ここで利用されているコマンドを知っていれば、機器に設定されたユーザー名やパスワードを取り出すことは難しくありません。このような方法を用いれば、初期パスワードを変更した機器のユーザー名とパスワードを取得できてしまいます。

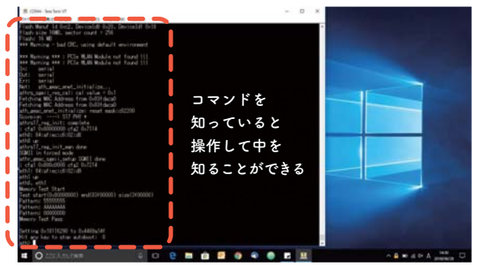


図8 機器上のプログラムを操作可能な状態で見ることができた

◆ つながる世界の開発方針

ここまでに述べた方法によって、インターネットに接続される機器のパスワードを盗み出し、機器を不正に動作させることが可能であるとわかりました。では、我々はどのようにしてハッカーから身を守ればよいのでしょうか。

独立行政法人情報処理推進機構（IPA）では、「つながる世界の開発指針」というガイドブックを用意しています。これに記載されている内容と今回の事象を照らし合わせると、以下の指針が参考になります。

［指針5］つながることによるリスクを想定する

この中の対応例にIoTコンポーネントとしてのリスク想定という項目があり、具体例として次のように記載されています。

① 出荷時の初期パスワードを同一にしない、また、推定されにくいものとする。
② ユーザ側でのパスワード変更を必須とし、パスワードの自動生成またはユーザが入力したパスワードの強化をチェックする。

［指針7］物理的なリスクを認識する

この中で、次のポイントについて注意すべきとされています。

① 盗まれたり紛失した機器の不正操作や管理者のいない場所での物理的な攻撃に対するリスクを想定する。
② 中古や廃棄された機器の情報などの読み出しやソフトウェアの書き換え・再販売などのリスクを想定する。

上記以外にも、さまざまな指針がガイドブックに記載されています。IoT機器のベンダーにおいては設計・製造時に、ユーザーにおいては利用時に、的確な形でガイドブックに記載された指針に沿って対策を見直すことで、セキュリティ面で不安の少ない安全な社会を築くことができると考えられます。

JTAGとは

JTAGはもともと、JTAGに基づくインターフェースを持ったデバイス同士の製造時の品質検査のために開発されたものです。このインターフェースはデバイスの細かな制御が可能であるため、マイクロプロセッサーの動作も細かに制御できるようなコマンドが追加されてきました。これを利用したものがJTAGテスター、JTAGデバッガーといわれるものです。

JTAGテスターは文字通り、JTAG規格で接続されたデバイスがプリント基板上に実装されたときに製造工程でデバイス間の結線が正しく行われているかをチェックするために利用されます。

JTAGデバッガーは、マイクロプロセッサーに対するデバッグコマンドをJTAG経由で送り、マイクロプロセッサー上のプログラムの動作をデバッグするために利用されます。その昔、ICE（インサーキット・エミュレーター）という機器が存在していました。ICEとJTAGデバッガーの違いは、ICEが基板上のマイクロプロセッサーの動作を全て取り替えて動作するのに対し、JTAGデバッガーは基板上のマイクロプロセッサーにコマンドを送って動作する点にあります。このため、ICEはプログラムが全く完成していない状態からデバッグできますが、JTAGデバッガーはマイクロプロセッサーがメモリの読み書きを正常に行える状態にならないと利用できません。

ARM CPUの今までとこれから

サイバー・グリッド・ジャパン
IoT技術研究所　リサーチャー　宮崎 力

ARMという会社をご存じでしょうか?

CPUコアの設計・ライセンスを行っている英国の会社で、恐らく皆さんが今お使いになっているスマートフォンにも、ARM CPUが搭載されているはずです。

ARM CPUが搭載されているのはスマートフォンだけではありません。プリンターなどのパソコン周辺機器やデジタルカメラ、お掃除ロボットやカーナビなど、今やIoT機器で最も多く使われているCPUはARMであるといっても過言ではありません。

ARMとの出会い

なぜ、1つのCPUアーキテクチャがこれほどまでに普及したのでしょうか?

私が初めてARM CPUに触れたのは2000年頃です。

当時勤めていた会社の社長が絵に描いたようなCPUオタクで「自分で設計製造したCPUボードを売って儲けるんだ」と常人には理解されないであろう夢を抱いて独立起業した方でした。その夢は当時の社員たちの協力のもとで実現し、最初に世に出したCPUボードでARMを採用したというわけです。

採用の理由は「なんとなくはやっているから」でした。当時は今ほどARM CPUが普及していたわけではなく、SuperHやPowerPC、MIPSなど多くの選択肢があったのですが「なんか最近よく名前聞くし、なんかかっこいいし、それを実装したCPUボードを出せば話題になるんじゃね?」と、社長を含め全員が20代という若いメンバーで話し合った結果、ARMの採用に至りました。

さて、実際ARM CPUボードでちまちまとプログラムを書いていた私でしたが、ずっと解消されない疑問を抱えたままでした。それは、なんでこのCPUがはやっているのだろう？　という疑問です。なにせマニュアルは全部英語で書かれていますし、似たようなスペックのMIPS CPUボードで同じベンチマークプログラムを走らせても、圧倒的にARMの方が遅かったからです。

ARMがブレークした理由

私にとっては実にとっつきにくいARMでしたが、当時話題になっていた要因の一つは、携帯電話大手Nokiaが、自社の携帯電話のCPUにARMの採用を決定したことでした。採用の理由は、他のCPUに比べ低消費電力だったから。当時はスマートフォンという言葉もなかった時代で、携帯電話に求められる処理性能はそれほどでもなかったのです。処理パワーが十分なCPUが複数あるのなら、その中で最も低消費電力なのはどれかと選べば、ARM以外にありませんでした。

かつて高性能CPUといえばMIPS、SPARC、PowerPCがその代名詞でしたが、ARMは初めからハイスペック競争には参加していませんでした。後発の彼らが、同じ土俵で横綱たちと戦っても勝てないと思ったのでしょう。サーバーやワークステーションという横綱たちと同じ土俵に上がることなく、ARMは組み込み分野に目を付けたのです。

そこで彼らが、自らのCPUのウリとして全面的に押し出したのが、低消費電力であること、です。

組み込み機器のCPUとして最も重視されるのは、CPUパワーが要求条件を満たしていること、次に、低消費電力であること、です。当時ハイスペック競争の真っただ中にあったRISC業界において、思い切った戦略だったと思います。

ARMの戦略は当たりました。

ちょうどその頃、自社の携帯電話に搭載すべく、バッテリー駆動が可能なCPUを探していたNokia社の目に留まり、採用に至ったのです。

これはARMがブレークするきっかけになりました。各メーカーが自社製品にARMを採用する事例が一気に増えたのです。採用の根拠をわかりやすくいうなら「なんとなくはやっているから」。ちょうどIBMに採用されたのをきっかけに、パソコン向けのCPUで世界一になったIntel CPUと同じことが、組み込みの分野でARM CPUにも起こったわけです。

ARMの戦略は偶然当たったのでしょうか？　私はそうは思いません。大手企業の決定には必ず後続が生まれます。意思決定の根拠は「なんとなくはやっているから」です。ですが、後続は何の考えもなしに流されてそうするわけではありません。ユーザーが多いほどモノとして枯れており、選択する上で合理的だからです。恐らくARMは、Nokiaがそのとき求めているものを注意深く観察し、確信を持って設計をしたに違いありません。

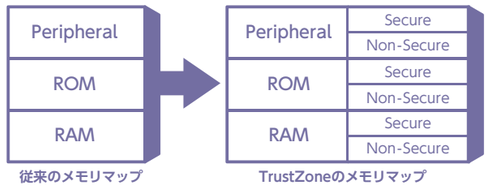
ARMの市場を見極める力と決断力は称賛に値するでしょう。

ARMのセキュリティ機能

そんなARMが今、自らのCPUのウリとして全面的に押し出しているのがTrustZoneと呼ばれるセキュリティ機能の拡張です。

前述したように、ARMは市場に求められているものを敏感に察知し、CPU設計に反映してきました。メモリリソースが限られているという声には、コード密度を高めるためThumbといわれる16ビット命令セットを追加しました。また、音声や動画を扱う際の処理を高速化したいという声には、SIMD(Single Instruction Multiple Data) といわれる命令セットを追加することで応えました。TrustZoneは、モバイル決済やIoT機器の普及に伴うセキュリティを重視する市場の声に応えたものといえるでしょう。

TrustZoneの特徴は、CPUメモリマップの各領域にSecureかNon-Secureかという属性を持たせたことです。



従来のメモリマップ　TrustZoneのメモリマップ

Secureな領域に配置されたプログラムコードはSecure/Non-Secureの領域いずれにもアクセス可能です。そして、Non-Secureな領域に配置されたプログラムコードは、Non-Secure領域にのみアクセス可能で、Secure属性を持つ領域にはアクセスできません。

十分にレビューされ信頼されたプログラムコードと機密性の高いデータをSecureな領域に配置し、それ以外のプログラムコードやデータはNon-Secureな領域に配置することで、Secureな領域への不正なアクセスを防ぐことができます。

仕組みは至ってシンプルですが、実装者視点に立ってみるとプログラムコードの配置に悩む仕組みでもあります。

例えば、ユーザーからの入力をトリガーに指紋データにアクセスしたい、というようなとき。

指紋データそのものや指紋データにアクセスするプログラムコードはSecureな領域に配置したとします。そして、ユーザーからの入力を受け付けるプログラムコードは、一般的なアプリケーションコードなので、Non-Secureな領域に配置したとします。

では、アプリケーションコードでユーザーからの入力を受け付け、いざ指紋データをSecureな領域に保存したいという場合どうするのでしょうか?　Non-Secureな領域にあるコードからSecure領域にアクセスすることになりますから、不正なアクセスとなりフォールトが発生します。

このため、Secure領域の一部はNSC(Non-Secure Callable)という属性を定義できるようになっています。ここに、Secure領域にアクセスするためのAPIテーブルを用意しておくのです。Non-Secureなアプリケーションコードから、Secureな領域にアクセスする場合は、NSC領域に定義されたAPIを経由することで、安全性を確保します。

ちょうど、プライベート変数を直接触るのではなく、プライベート変数にアクセスするためのAPIを用意し、そのAPIを介して変数にアクセスすることで安全性を確保するプログラミングの考え方と同様です。

ARMのセキュリティ機能をもう少し紹介します。

最近の組み込み向けCPUには、暗号エンジンが搭載されているものが増えました。暗号処理はソフトウエアだけでもできますが、暗号鍵の保存方法が悩みどころですし、処理速度やメモリリソースに問題が出ます。これをハードウエアがサポートすることで、セキュリティと高速化と省メモリを同時に成立させようとするアプローチです。ARMからはCryptoCellというセキュリティIPコアが提供されています。

ただ、暗号化というのは基本的に通信の秘匿を目的としたものです。確かに不正アクセスはネットワーク経由で行われるものが大半ですが、物理的なアクセスには弱いという面もあります。例えば、デバッガを使用したリバースエンジニアリングです。

Secureな領域に配置した秘匿性の高いプログラムコードであっても、CPUのデバッグポートに物理的にデバッガを接続してしまえば、リバースエンジニアリングされる可能性があります。このような事態を想定し、XOM(eXecute Only Memory)という機能が搭載されました。

これは、実行のみを許可するメモリ領域であり、デバッガですらリードもライトもできません。通常デバッガは命令コードをロード命令で読み、それを逆アセンブルすることで処理内容を認識しますが、XOMメモリではそれが不可能になるということです。

ARMのこれから

我が世の春を謳歌していたかに見えたARMですが、2016年にとんでもないニュースが入ってきました。なんと日本の企業がARMを買収したというのです。その額、3.3兆円。その企業の名は、ソフトバンク。

買収の目的は何なのか、さまざまな憶測が飛び交いましたが、新たにARMの会長に就任した孫正義氏が強調していたのが、IoTにおけるARMのシェアとセキュリティです。

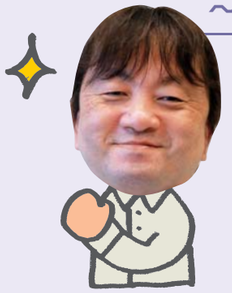
彼らの動向から目が離せませんが、間違いなくIoT機器におけるセキュリティガイドラインの策定に動くはずです。圧倒的なシェアを手中にした彼らが決めたルールであるなら、IoT機器に携わるその他の者は追従せざるを得ません。これはハード・ソフトを問わず、強い影響力を持ったものになるでしょう。

この事態は、エンドユーザーにとっては喜ばしいことです。あらゆるものがインターネットに接続され、それらの機器を安心して使えるのは理想的ともいえます。しかし、そのガイドラインへの追従が多大なコストを要するものだとしたら……。個人的には、組み込み分野における半導体メーカーをはじめとした企業の勢力図にも注視したいところです。

Youは何しにラックへ？

Why did you come to LAC?

～IoT技術研究所のスタッフ2人の素顔に迫る～



2017年11月の設立から10カ月を迎えたラックのIoT技術研究所。バズワードとして定着し、今をときめく先端技術でもある「IoT」の研究とはさぞ華やかなものだろう、と思いきや、仕事の内容は意外にも「地味」なのだとか。
そんな「地味」な作業の中に潜む魅力とは？ そして、そんな「地味」な作業をするために、彼らはなぜラックにやってきたのか？ IoT技術研究所のスタッフに、話を聞きました！

ラックにIoT技術研究所ができた背景とは？

木田：もともとはIoTチームという名の下で業務を行っていたんです。ようやく人が揃ってきたので、研究所という形にしようとなり、発足したのが2017年11月1日。現在は、所長の渥美と僕、宮崎の3人で運営しています。

宮崎：僕がラックに入社したのは2017年12月なので、実質的には技術研究所が立ち上がってから加わった形になりますが。

木田：実は宮崎とは前職が同じ会社で。ラックの社員で頻繁に勉強会を開催している人がいて、そこに宮崎を誘って参加していたことがきっかけでまず僕がラックに入り、そのあと宮崎を呼び寄せたという経緯です。

お二人は前職ではどんな仕事を？

宮崎：僕はずっと組み込みのソフトウエアエンジニアをやっていました。OSといえばWindowsが一番有名ですが、僕はWindowsの開発はほとんどやらず、LinuxやTRONといった、今でいうところのIoT機器の中で動作するOS上での開発をメインにしていました。Windowsが組み込まれるような高機能な機器の開発に関わる機会があまりなかったというのもありますが、まあ、Microsoftが決めた仕様に追従するのが大変そうだというか（笑）。もっとも、Microsoftが嫌いというわけではなく、むしろ好きなんですけれど。

木田：僕は、最初はハードウエアエンジニアだったんですけど、会社から突然「ソフトウエアを作る人間がいないからやって」といわれて（笑）、そこからソフトウエアエンジニアに転向しました。おかげさまでIoTの基礎となるハード、ソフト、両方の知識が身について、ラックに入る前まではずっとIoT機器の開発をしていました。

IoTの技術研究の魅力って何ですか？

木田：IoTってまだ仕様がきっちりと決まっていないので、自分の意思をシステムに反映しやすいんです。その代わり、中身が全く見えない。

宮崎：「見えない」というのは、まさにIoTの醍醐味ですね。「プログラマー」というとパソコンの中で動くソフトを作るイメージがあると思うんですけど、IoTの場合はソフトを動かす場所がパソコンではなくて機器の中になってくる。そうなると、パソコンの画面では見ることができないんですよね。見えないんですが、なんというか、やっているうちに見えてくるというか。その感覚は楽しいです。

木田：IoTには「画面」がないことが多いので、システムが設計した通りにできているかを特殊な機械で一つひとつ確認するのですが、この作業が地味に長くて（笑）。見えないだけに、一人で考えていると行き詰まることもありますし。そういうときは勉強会で人の意見を聞いてみると、視野が広がったりします。

宮崎：木田は普段から興味を持った勉強会に積極的に参加していますから。僕は誘われたときくらいしか行かないですけど。休みの日は休みたいっていうか、そっちの欲が勝ちますね。

木田：そうか……。今まで僕は無理に誘ってたのか。申し訳ない（笑）。

現在は、オートモーティブ関連の研究をしているそうですね？

宮崎：はい。つい最近始まった案件で、まだ軌道にも乗っていない状況なんですけど、でも楽しいですね。個人的に車やバイクをいじるのが好きだということもあって。

2017年6月入社。IoTという言葉がない頃から組み込みシステムの開発を行う。ハードウエアエンジニアからソフトウエアエンジニアに転向した異色（？）の経歴の持ち主。休日も技術情報をかき集める根っからの技術好き。技術から離れたところでは、CDショップに入り浸り、ヨーロピアンジャズに没頭するのが常。

木田 良一
Ryoichi Kida
サイバー・グリッド・ジャパン
IoT技術研究所
リサーチャー

宮崎 力
Tsutomu Miyazaki
サイバー・グリッド・ジャパン
IoT技術研究所
リサーチャー

2017年12月入社。LinuxやTRONといったプラットフォーム上で動くデバイスドライバやアプリケーションの開発などをおよそ20年にわたり行ってきた、ソフトウエアエンジニア。休日はもっぱら、バイクと車に時間を費やし、ドライブやチューニングにいそしむ。

木田：自分たちで本当にできるのか？　という手探り状態からのスタートだったのですが、半年ほどたってようやく自分たちにもできそうだと感じ始めたところです。なにせ「見えない」ので、IoTのシステム開発はアイデアや雑談から始まることが多いんです。

宮崎：だから、僕らの仕事は「遊び」の時間が大事なんですよ。「いつまでにこれを完成させる」といったようないわゆる「お仕事」的な作業が新しいアイデアに結び付くことは正直あまりないんですけど、自宅で、「こんな技術が出てきたんだ」「こんな風に触ったらどんな動きをするんだろう」とパソコンをいじりながら得る情報や学びっていうのは絶対に無駄にならないなと思っています。後々役立ててやろうと思ってやっているわけじゃないんですけど、ふとしたときに「あのときのこれだ」とピンとつながる、そういう感覚がありますね。

とても楽しいお仕事のようですが、逆に大変なことってありますか？

木田：苦労ももちろんありますよ！　IoT技術研究所が立ち上がるまで、ラックはIoTの研究開発に必要な機材も何もない会社でしたから、まずはIoT機器を設計するための環境作りからスタートしなければならなくて。

宮崎：僕が入社したときには組織自体はできていましたが、どうやって仕事を進めていくか、いわゆるプロジェクト管理のスタイルができあがっていませんでした。だから、話し合いの繰り返しでしたね。「今まで自分はこういうツールを使ってきたから、ここでもそれを導入してみたらどうか」といったような細かいところから決めていきました。

木田：それから、今まではお客さまの要望に応じて設計だけをしていればよかったのが、研究職だとそうはいかない。いろいろな人の意見を取り入れていかなければならないので、人の話を聞く能力や、人に分かりやすく説明する能力が必要です。一緒に仕事をする社内外のスタッフに、自分の考えを伝えて仕事を作り上げていくというのが一番大変ですね。

宮崎：今まではお客さまからいただいた仕事をこなすことで会社に対して利益を出していたわけですが、研究という仕事はそれがお金になるかどうかが見えない中で進めていくことになります。だから、自分がやっていることをどうやってお金に変えていくかということを自分なりに考えなければいけない。それができなかった場合はどうなるんだろう、なんていうことをぼんやり考えたりすることもあります。

木田：企業として視野や見識を広げていくことは成功可否にかかわらず必要なことなので、僕はそのあたりはあまり気にしていません。ラックでこういう研究をやっている、そのことをお客さまや外部の方に知っていただき、相談できるパートナーとして認知されれば必然的にお金につながっていくのかな、と思います。

最後に、IoT技術開発に向いている人って、どんな人でしょうか？

木田：仕組みに興味があって、その仕組み自体を自分で作ってみたいとか変えてみたい、そんな人ですかね。好奇心が旺盛で、そしてそれを維持できる、若い人かな。

宮崎：IoT機器のベースは「マイコン」といって僕らより少し上の世代が触れてきたものにあるんですね。だから、若い人が自力で勉強するのはなかなか難しい。そのあたりは僕らがサポートするので、何を取っ掛かりにしたらいいか分からないけど興味だけはある、そんな人がいれば、ぜひ来てほしいですね。

ラックの顔

第6回

技術書の翻訳を通じて考える自動車セキュリティに対してすべきこと

自動車のセキュリティに関する技術書『カー・ハッカーズ・ハンドブック』。この本は、車両のセキュリティに関する研究者Craig Smith氏が執筆した『The Car Hacker's Handbook』の日本語版で、翻訳を行った“謎の団体”「自動車ハッククラブ」のメンバー11人中、9人がラックグループの関係者だ。日常業務とは色を異にする「書籍の翻訳」を行うに至った経緯と苦労を、自動車ハッククラブのメンバー2人が明かした。

以前から「自動車のセキュリティ」に対して関心を持っていた二人

『カー・ハッカーズ・ハンドブック』とラックとの接点を作ったのは、IoT技術研究所の所長を務める渥美清隆だ。2016年3月に発売された原書『The Car Hacker's Handbook』がセキュリティ関係者の間で話題となっており、興味を持った渥美が同書を購入したのが2016年7月のこと。「口語やスラングを混じえつつも、おおむね平易な英語で書かれていて読みやすい」と感じた渥美が、同書をテキストに勉強会を開催することにし、イントラネット上の社内掲示板で参加者を募ったことが訳書出版の原点だ。

「普段からできる限り社内の勉強会には参加するようにしている」という金子と北原だが、「自動車のセキュリティ」というこの本のテーマには、ことさらに関心を持ったと二人はいう。「コンピューターがサイバー攻撃を受け、突然街を破壊しだす――。今はまだ漫画や映画の中の話に過ぎませんが、そんな風にサイバーの世界で起こったことが現実の世界に影響を及ぼすようになったら危険だな、というようなことは以前から考えていました。一般的で身近で、機械であることを特段意識せずに人が動かしていて、そして最も整備が進んでいるインフラである“自動車”がサイバー攻撃の対象になったらどうなるか。そんな興味があって、昔から自動車のセキュリティについては個人的に調べたりしていました（金子）」

一方の北原は、次のように話す。「自動車のセキュリティは、一般的なサイバー攻撃とはだいぶ趣が違うと以前から感じていました。実際に回路を組んで模型を作り、解析を行うというようなハードウエアの知識も必要になってくるので、ソフトウエアの技術者が活躍する一般的なIoT技術と異なり、ハードウエアの技術者の力量にセキュリティの担保の精度が大きく左右される。メーカーや車種によって通信の内容も変わってきますから、ある程度体系化できる従来のコンピューターのセキュリティとは一線を画していると思います（北原）」

最終的に、勉強会には渥美のほかに14人のメンバーが集まった。『The Car Hacker's Handbook』は13の章と付録によって構成されていたため、勉強会は、全体を14に分けて各人が業務内容や素養に合わせて担当の章を持ち、プレゼンテーションを行う輪講のスタイルを取ることとなった。勉強会がスタートしたのは、2016年9月。およそ2週間に1度のペースで、11月まで行われた。

個性豊かな勉強会メンバーの発表が訳書の発行へと発展

勉強会を始めた当初は、メンバーで翻訳をやることになろうとは思っていなかったと二人は話す。「出版社に話を持ちかけてみてもいいかもしれない、という程度の空気はありましたが、自分たちで翻訳をするという前提ではありませんでした（北原）」

「そもそも、勉強会のメンバーが全員、原書を隅から隅まで読んでいたわけではありません。私の担当は6章でしたが、当然のことながら、6章は1章から5章までの内容を踏まえたものになっています。分からない単語にぶつかるたびに前の章に戻り、ということを繰り返していたので、結局は読まざるを得なかったわけですが、中には前段を参照せずとも発表ができるような章もありました。また、発表の仕方にもかなり個人差がありました。画像をたくさん使った、手が込んだ資料を作ってくる人もいれば、メモ帳で発表をする人もいるといった具合でしたから。（笑）（金子）」

そういった「フリースタイル」の勉強会が翻訳チームの結成へと至ることになった契機は、勉強会の主催者である渥美の発言にある。「せっかくここまで勉強したんだから、ラックで訳書を出してみよう」という渥美の提案に、多くのメンバーが賛同。渥美がつながりを持っていた広島市立大学の井上博之准教授（情報工学専攻）に監修を依頼し、訳書発行のプロジェクトが始動することとなった。

行間を読み解き、分かりやすい日本語に置き換える「翻訳」の難しさ

翻訳は基本的に、それぞれが勉強会で受け持った章を訳すというスタイルとなった。一度、発表のために目を通していたとはいえ、訳文を作るという作業には、また別の苦労があったと二人はいう。語学能力に長け、2014年から情報セキュリティ国際会議「CODE BLUE」の翻訳スタッフに従事する北原は、こう話す。「私が訳した11章は、顧客のネットワーク環境に擬似的なサイバー攻撃をかけて脆弱性診断を行う、という日常業務の知識を生かせる内容でした。そのため、英文を読み解くことに関してはさほど労力を必要とはしなかったのですが、英語をそのまま正しい日本語に成形すれば訳文になるわけではありません。“分かりやすい日本語”の文章を作るため、できるだけ原文の意味を損ねずに日本人になじみのある言葉に置き換えるという作業には難しさがありました。例えば、私は11章の英文のタイトル“Weaponizing CAN Findings”を“攻撃ツールの作成”と訳しました。どの言葉を選ぶかというのは本人の知識量やセンスに依存しますので、そこは皆それぞれに苦労したのではないでしょうか（北原）」

片や金子は、最も苦心した点として「行間を読み解き、書き下すこと」を挙げる。「著者は車載システムに関して素養がある方ですし、そもそもいわゆる一般書ではありま

金子博一
hirokazu kaneko

北原　憲
ken kitahara

せんので、ある程度知識を持った人が読むことを前提に書かれている本ではありました。そのため、“知っていて当然”という感じで内容が省略されているような箇所が結構ありましたね。勉強会のときもそうでしたが、そういう箇所を見つけたらできるだけ裏を取って補足するよう心がけていました。私は普段、セキュリティアカデミーというセクションで、マルウエアの解析について人に教える仕事をしています。仕事柄“人に分かりやすく伝えるためにはどうしたらいいか”を常に考えていますから、そういう意味では日常業務が今回のプロジェクトに生きたかな、とは思います（金子）」

ハッククラブのメンバー全員が、それぞれに苦労をしながら完成させた訳文は、さらに相互レビューで全体のトーンや文言の統一などを図った上で監修の井上准教授の手によってブラッシュアップされ、2017年12月、株式会社オライリー・ジャパンより『カー・ハッカーズ・ハンドブック』として発行された。同社の担当者によると、本の売れ行きは好調であるという。

専門化するセキュリティ業界の中で自動車のセキュリティに挑む

今回、翻訳を通じて『The Car Hacker's Handbook』を読んだことで、金子には新たな気付きがあったという。「昔の自動車は、アクセルを踏むというアクションがそのままエンジンの動作に働きかけるという回路になっていたのですが、今はそうなってはいません。アクセルを踏むと、その情報が信号としてエンジンに伝わる仕組みになっているので、実際にアクセルを踏んだかどうかということは関係がなく、信号が届いているか否かが重要になっているんです。自動車という機械は絶対に誤作動が許されないものですから、私はてっきりドライバーの動きが物理的に動作に関わっているのだろうと思っていたのですが、そういった部分もすでに通信に置き換わっている。そこに私は驚きを隠せませんでした。当然のことながら、その回路にはいろいろなプロテクトが働いているわけですが、それを解析することで、ひょっとしたら攻撃者側が何かをできる、そういう部分が見えてくる可能性もあるのではないかと。まさに、私が想像していた“映画の世界”が現実になるときがくるかもしれない、という気になりました（金子）」

信号処理に関する記述の多さには北原も同様に驚きがあったという。その上で、北原は「セキュリティ会社として自動車のセキュリティにどう関わっていくべきか」について改めて考えさせられたと話す。「まずは車の構造をきちんと理解することが重要だと思います。その上で、攻撃者がどういったところからどういう攻撃ができるかを網羅し、自動車の中で動いているファームウエアの脆弱性を探す。初めにお話ししたように、ハードウエアの知識が必要な分野なので、我々のようにソフトウエアを専門とする技術者ではどうしても踏み込めない領域というものがあります。ソフトウエアの技術者とハードウエアの技術者とが、今まで以上に手を取り合っていくことが求められていると思います（北原）」

重ねて北原は、今後セキュリティ業界にはさまざまな専門分野ができていくだろうと予測する。「私は物理学科出身なので物理学にたとえますが、私が大学でやっていた光の研究のほかにも、物理学には半導体、素粒子、原子核、宇宙などさまざまな専門分野があります。IoT機器の参入により、情報セキュリティはすでに一人の手には負えないものになっています。そういう意味で、セキュリティもだんだんと専門分野を持つ“学問”に近いものになっていくのではないかと感じています（北原）」

今後、自動車のIoT技術開発が進み、コネクテッドカーが実現するようになった場合、ラックはこの分野にどういったアプローチをしていけばよいのか。金子はこのように考えているという。「自動車は“走るコンピューター”ですから、診断の技術や攻撃を見分ける技術といった当社の知見は大いに生かせると思っています。また、コネクテッドカーの最大の利点は、ユーザーにさまざまな情報を提供できること。そうなると、コネクテッドカーだけでなく、例えばユーザーにリアルタイムで渋滞の情報を伝えるアプリといったような、コネクテッドカーを支える周りの環境も含めてうまく回していく必要があります。ラックが積極的にリードを取り、業界全体で自動車のセキュリティを健全なものにしていけたらいいですね（金子）」

できるだけ多くの人に情報発信を行い、業界を盛り上げたい

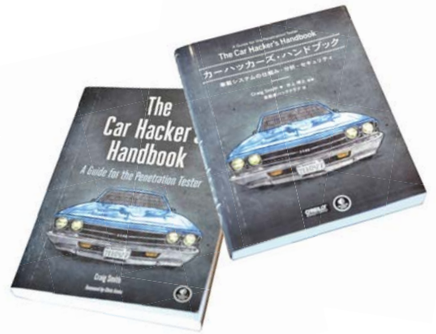
今回の「自動車ハッククラブ」の活動のように、業務とは直接関係のない課外活動にも積極的に手を挙げている金子と北原。特に北原は、社内のみならず社外の人間との交流もアクティブに実践している。「最近は、同じようなことを専門にしている技術者同士で情報交換を行うコミュニティを立ち上げ、運営しています。より多くの人を巻き込んでいくためには、自分の利益だけを考えるのではなく、相手にきちんと貢献できる組織であることが重要。日頃から新しい技術的な話題などを収拾し、率先して情報発信を行うことでまずは相手に認めてもらい、自分との付き合いの結果として得られるものがあるということをきちんとアピールすることを意識するようにしています（北原）」

一方、自らを「控えめな人間」と評価する金子は、自分がリーダーシップを取って何かを進めようということまではあまり思っていない、と笑う。「とはいえ、考えてみると業務上で何か新しいことが始まる際にメンバーとしてアサインされたり、プロジェクトのリーダーを任されたりという場面は少なくないかもしれません。それは、周囲の人の信頼があるからだとポジティブに考えるようにしています（金子）」

それぞれに違ったスタンスで課外活動に臨んでいる二人。入社して丸10年が経ち、「中堅」の域に入った金子は、若くして積極的に活動する北原に対して次のようなアドバイスを贈る。「北原君のように先陣を切って走る人、そういう人に認められた人はいろいろな方面から声がかかりやすい。今の技術者の中にはそんな現状があるように思います。ただ、日本人には自分の能力ややる気を表に出すことが苦手な人も少なくありません。当然、アピールが上手な人は優先的な待遇を受けて然るべきなのですが、そうでない人でもいろいろな活動に参加できるよう、できるだけフラットな視点で告知をしてもらえたらいいなと思います（金子）」

一方、入社5年目を迎えた北原も、より多くの人に情報発信を行えるような仕組みを画策しているという。「技術について配信するメールマガジンのようなものを2年ほど前から発信していますが、そういったところで勉強会の告知をするなど、社内の幅広いところに情報を拡散できる体制を作っています。業務の傍らなので去年はあまり活動できませんでしたが、今年はもっと回数を増やしていけたらいいなと思います（北原）」

日常業務では全く接点を持たないという二人が、課外活動という場を通じて意見交換を行う。こういったコミュニケーションの延長上に、ラックの将来を担う人材が生まれるのかもしれない。



金子博一
hirokazu kaneko

サイバーセキュリティ事業部
セキュリティコンサルティング部
セキュリティアカデミー

2008年ラック入社。サイバー・グリッド研究所にてマルウエア関連の研究に従事。本誌などを通じ、マルウエア解析の統計情報などを開示。現在はセキュリティアカデミーに所属し、主にマルウエア解析能力を持つ人材育成に携わっている。2歳の双子の父親でもあり、子煩悩な一面も持つ。子の健康を考慮した料理に力を入れ始めたはずが迷走を開始し、最近は皮から水餃子を作ることにはまっている。

北原憲
ken kitahara

サイバーセキュリティ事業部
セキュリティ診断部 ペンテスト技術グループ

2014年ラック入社。物理学で博士号を取得後、入社に伴い情報セキュリティ技術者に転向。現在はラックを代表するチーフペネトレーションテスターとして、新規サービスの考案や技術開発を始め、Hardening ProjectやCODE BLUEなどのイベント運営や、OWASP Japanのようなコミュニティの運営、各種執筆活動に携わっている。英語のほか8言語を操る多彩な語学力の持ち主。

巻末あとがき

執行役員　サイバー・グリッド・ジャパン GM　加藤 智巳

1988年、アメリカで商用インターネットが始まり、同時に日本でWIDEプロジェクトが開始した頃、私はまだインターネットの存在を知らずに、社長と自分だけしかいないシステムハウスで、家電メーカーから請け負った製造ラインの試験工程を自動化する仕事に没頭していました。ライン作業員による測定機器の目視で行っていた検査を、はやりの16Bitパソコンを駆使して、GPIB経由で測定機器を制御できるようにするというものなのですが、試験対象の基盤のデータを取るためには、インターフェースを自作しなければなりませんでした。そのラインにはさまざまな業者が請け負いで出入りしていましたが、ソフトを請け負う業社はハードが分からず、ハードを請け負う業社はソフトが理解できないといった状況だったので、私たちはソフトとハードを同時に請け負うことができる企業として大いに重宝されていた記憶があります。

その後、ネットワークSIの会社に転職し、LANのエンジニアになった時は、OSI参照モデルの1～4層のことだけを意識すればよく、ある意味、アーキテクチャ的に分業化された業務内容に当時の情報システムの先進性のようなものを感じていました。

そのうちにネットワークシステムが「インフラ」という一言でくくられてしまい、ただの土管のような扱いとなった頃には、現在のIoTセキュリティで議論されるさまざまな課題など想像もつかなかったと思い返す今日この頃です。

私が2014年にサイバー・グリッド・ジャパン サイバー・グリッド研究所長に就任した頃、スタッフにIoTセキュリティを研究させたいと考えても、組み込み系のシステムが分かるエンジニアは2人しかおらず（しかもそのうちの1人は趣味で）、ハンダごてを使える者が圧倒的に少ない状態でした。何から始めたらいいのかも分からない中、とりあえず少ない研究費でネットワーク機能のついた家電品を片っ端から買い漁ってイジりまくれと指示したのが、当社がこのテーマに取り組んだ最初の一歩だったように思います。その後研究環境を少しずつ整え、当社は2017年にIoT技術研究所の設立を果たしました。

今号に執筆したリサーチャーも語ってくれていますが、IoTセキュリティはハード、ソフト、さらに運用管理、標準化、IoT利用市場と幅広い視点で研究しなければならない重要なテーマです。ようやく手応えを感じ始めた今、これからも柔軟に粘り強く研究を継続していきたいと強く思っています。

CYBER GRID JOURNAL VOL.6

サイバー・グリッド・ジャパンは株式会社ラックの研究開発部門です。
サイバー攻撃や各国のセキュリティ事情、セキュリティ防御技術などに関する最先端の研究のほか、
複数のセキュリティ企業との連携や新たな製品・サービスの開発、各種啓発活動などにより
日本のセキュリティレベルと情報モラルの向上に貢献しています。

株式会社ラック サイバー・グリッド・ジャパン
〒102-0093 東京都千代田区平河町 2-16-1 平河町森タワー
E-MAIL：sales@lac.co.jp　https://www.lac.co.jp/

株式会社ラック
サイバー・グリッド・ジャパン