目次

量子鍵リース

この記事では鍵データのリースを紹介します。「リース」は貸借といった意味ですが、鍵データの貸し借りは物理的な物体の貸し借りと同じようには実現できません。ここでは、鍵データを相手に渡して貸し出しの代わりとし、渡した鍵データを削除してもらうことで返却の代わりとします。

しかし、単に貸し出した鍵データそのものを削除させたとしても、その鍵データが複製されていないことを保証できません。よって、何らかの方法でデータが複製されていないことと、データを削除したことを証明しなくてはなりません。

今回紹介する量子鍵リース（Quantum Key Leasing）は、量子データの特性を使ってこれを実現します。量子データは重ね合わせを使って複数のデータを同時に持つことができます。これを巧妙に使うことで、重ね合わせられた状態であれば復号ができる鍵として機能し、重ね合わせ状態を壊した古典的なデータが削除の証明となります。もし借主が重ね合わせられた量子データから古典的な鍵データを取得するために不正な測定を行った場合は、削除の証明となるデータを作れなくなります。さらに重ね合わせ状態が失われたデータは鍵として機能しなくなります。

この記事では量子コンピュータの基礎的な知識を前提として、量子鍵リースを用いて古典コンピュータの公開鍵暗号の秘密鍵を貸し出す方法について[1]を参考に解説します。

1 古典計算と量子計算

量子鍵リースをはじめとした量子計算の議論に移る前に、まず前提として従来のコンピュータ（古典コンピュータ）による古典計算との関係について説明します。古典コンピュータ上のあらゆる計算は、最終的には回路で表現でき、その回路はNANDゲートのみから構成できます。ここで、もしNANDゲートを量子コンピュータの論理回路である量子ゲートで構成できれば、古典コンピュータ上のあらゆる計算を量子コンピュータ上で理論上は動作させられます。

そのために式（1）の行列CCXで表されるToffoliゲート[*1]を導入します。Toffoliゲートは3つの量子ビット$|a\rangle, |b\rangle, |c\rangle$を受け取って下記のように変換します。

$$\mathrm{CCX} \equiv \begin{pmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&1&0 \end{pmatrix} \tag{1}$$

$$\mathrm{CCX}(|a\rangle|b\rangle|c\rangle) = |a\rangle|b\rangle|c \oplus (a \wedge b)\rangle \tag{2}$$

式（2）より、Toffoliゲートはa, bを入力されたそのままの状態で返し、3量子ビット目は$c \oplus (a \wedge b)$の計算を行った結果になります。ここで$c = 1$に固定した場合、3量子ビット目のXOR計算は$1 \oplus (a \wedge b) = \neg(a \wedge b)$となり、つまり3量子ビット目は$a, b$を入力としたNANDゲートの結果と等しくなります。このことから、Toffoliゲートを用いればNANDゲートを量子コンピュータ上で実装することが部分的にできると言えます。したがって、理想的な量子コンピュータ上では、全ての古典コンピュータ上の計算を実行する能力があります。

一方で、Toffoliゲートをはじめとした測定を含まないユニタリな量子ゲートによる計算においては、古典コンピュータのNANDゲートのように2入力から1出力というような入力に対して出力が減るような計算はできません。量子計算の要請として、測定を除く全ての計算は逆計算が常に可能である必要があり、たとえばNANDゲートの1ビットの出力からでは、2ビットの入力が曖昧になってしまい復元できません。よってToffoliゲートは、計算結果に追加で元々の入力a, bもそのまま出力するようにしています。

このように、可逆計算が可能なように工夫する必要があるという点に注意すれば量子コンピュータ上でも結果として古典計算と同じような計算ができます。

---

*1 トフォリゲートと読みます。

超かぐや姫！に見るセキュリティ

1　はじめに

Netflix・劇場で公開中の映画「超かぐや姫！」は 2030 年の近未来を舞台とする SF 作品であり、その作中には仮想空間や AI など情報科学に関連する要素が数多く登場します。この記事では「超かぐや姫！」に完全にやられている筆者が、作品中から情報セキュリティに関連する部分をピックアップし、作品の考察を交えつつセキュリティ的な背景を解説します。

2　注意

この記事は「超かぐや姫！」の重要なネタバレを含みます。

また、この記事では評論のために作中のセリフや画面上に映る文字を一部引用します。引用内容は筆者が自身の文責で書き起こしたものです。

（次ページ以降、ネタバレを含むため、未視聴の方はご注意ください。）

ネルモードドライバによる監視は強制力が強く、このアンチチートシステムは強力なものと言えます。 今日、LoL 以外の人気オンラインゲームでも同様に強力なアンチチートシステムが導入されているケースは珍しくないことから、KASSEN のアンチチートシステムも同様に強力なものであると推測できます。 このような強力なアンチチートシステムを回避してチートを行う手法もありますが、コンピューターの基板上の特殊なプログラムを書き換えたり[*3]、コンピューターに特殊な機器を接続する[*4]といったやや面倒な手法が時に必要となります。 ですから、乃依の「めんど〜」という発言は本心から出たものだろうと推察できますし、乃依のバックグラウンドとしてハッキングスキルがある可能性も考えられるでしょう。

3.3　チートプログラムと無敵状態

冒頭の引用で示したように、作中で一瞬、帝が使用するチートプログラムが画面に映り込む場面があります。 このプログラムは C 言語で書かれたもので、内容からキャラクターの HP・攻撃力・守備力を無限大へ書き換えるものであることが分かります。映り込んだタイミングと帝の顔に浮かぶ赤い紋様から、このプログラムは帝が月人と近接距離で交戦する直前に発動しており、埒が空かないと判断した帝が腹を括り月人に対して無敵状態を発動するためのものであったと考えられます。

このプログラムは、かなり簡略化されてはいますが、実際のチートプログラムのエッセンスをよく捉えているように見えます。 筆者は、ゲームチートに対する防御を助言する立場から、ゲームチートについての調査・解析を行っていたことがありました。このチートプログラムは「キャラクターの状態を保持している構造体を参照し、その中に浮動小数点で保持されているステータス値を `INFINITY` へ書き換える」というような内容で、実際のチートプログラムも本質的にはこのような処理を行うことがあります。ただし、HP・攻撃力・守備力を無限大へ書き換えて無敵状態になるチートというのは監視側からすると分かりやすく異常な振る舞いとなるため不正検知をされやすいものとなり、実際作中で帝は直後にチートを検知されて警告を受けています。 これは、なりふり構わずにかぐやを守ろうとする帝の姿勢の表れとも言えるかもしれません。

---

*2　このような挙動はマルウェア（コンピューターウィルス）と紙一重であるとして問題視する声は以前から根強くあり、筆者としても同意するところです。

*3　Firmware reprogramming

*4　DMA Attack