Security Analysis of PassRules™

Jeff Diamond Ph.D.

TRLabs

August 2, 2013

Abstract

PassRules™is a patented protocol for user authentication which makes use of a dynamic password scheme. Instead of a static password, a user is assigned a sequence of simple subrules, each of which maps a grid of decimal digits called a challenge table to a single digit. In order to authenticate, a user is provided with a challenge table and he must respond with a dynamic Personal Identification Number (PIN) consisting of the images of the challenge table under the sequence of sub-rules which compose his pasrule. Unlike static password schemes, this provides some security even after an adversary observes a small number of these challenge-response pairs. This report describes the results of simulations executed to determine how the level of security, represented by the entropy associated with the adversaries uncertainty in guessing the passrule or PIN, varies as a function of the number of observations of challenge-response pairs. We investigate the use of multiple dummy tables and of restricting passrules to a set with less entropy leakage and we also outline a concept which can be used to combat phishing attacks. We also also investigate the effect of including or omitting constraints on the allowed passrules which are meant to enhance the memorability of the rules.

1 Introduction

We consider a dynamic password scheme called PassRule™, in which a user is supplied with a six by six grid or challenge table of decimal digits. In order to authenticate, the user must apply a passrule to the challenge table and respond with the resulting four digit Personal Identification Number (PIN). The passrule consists of a sequence of four subrules, each of which applies a simple function either to the digit in a particular cell in the table or to pair of digits contained in a particular pair of cells of the table. There are thus two types of subrules: unary subrules, which are applied to the contents of a single cell, and binary subrules, which are applied to the contents of a pair of cells in the table.

This scheme can be generalized to include a larger challenge table, multiple challenge tables, where the passrule is applied to only one of the tables, or passrules with more than four subrules. We will however, restrict ourselves at the outset to the current case with passrules of length four and a single six by six challenge table.

We also restrict ourselves to a subset of possible six by six challenge tables and to a subset of all possible passrules. In [3], Weber considers a similar scheme to passrules, called GrIDsure, where, instead of applying subrules to the contents of cells, the response to a challenge table is derived from simply copying the contents of a number of specific cells to obtain a PIN. Weber points out that, if the contents of each cell in a challenge table are chosen independently and uniformly at random, the adversaries chances of guessing a PIN can be improved by guessing a random cell sequence and reading the corresponding PIN instead of directly guessing a PIN. Weber proposed the the use of grid balancing to address this problem, where the challenge tables are chosen so that the number of occurrences of each digit in the table is as close as possible to the same number for all other digits. In the case of six by six tables, this means that for each table, four of the ten digits appear three times each and six of the ten digits appear four times each. This scheme is also adopted by

PassRules™, and so we restrict our attention to challenge tables thus formed.

Subrules are restricted to a set of fifteen simple operators. Ten of these are unary operators, and they simply choose one cell from the table and add a fixed digit $d \in \{0, 1, \ldots, 9\}$ to the contents of that cell modulo ten. The other five allowed operators are binary operators and operate on the contents of a pair of cells by calculating the sum, difference, product, minimum or maximum of the contents of the two cells. The results are expressed modulo ten. A combination of an operator and a choice of cell or cells yields a subrule and a sequence of subrules is a passrule.

We refer to the ordered list of operators used in the subrules for a passrule as an operator sequence. In addition to restrictions on the challenge tables and the operators used, the default PassRule scheme includes three restrictions on the form of the operator sequences which are allowed. This is in order to ensure that the passrule is easy to remember and easy to apply. Passrules cannot reuse cells, passrules must use at most two distinct operators and passrules must have at most two transitions between operators. This means that a passrule of the form ABBA would be allowed, whereas one of the form ABAB would not. We focus, for the most part, on rules which satisfy these constraints, however we also consider the effect of omitting the constraints.

In [2], the entropies are calculated for passrule schemes with four and five sub-rules. This assumes that cells and operators are chosen uniformly at random, so that the entropy is simply the logarithm of the number of possible passrules. This entropy corresponds to the adversaries uncertainty in guessing the passrule before observations of any challenge-response pairs. The author goes on to describe how PINs or passrules can be guessed by an adversary which has the ability to introduce his own challenge tables and receive responses from the user. This introduces the threat of phishing, which is a common ploy used to obtain passwords and other private information from users who are redirected, for example often by an email, to a website which is masquerading as a trusted site.

A discussion is included in [1] and in [2] on entropy leakage in the case where the adversary is a passive observer but has access to some number of challenge-response pairs from a user. In this case, if a secret passrule r is chosen uniformly at random from a list of N possible passrules, the entropy associated with the adversaries uncertainty of the passrule is initially given by $\log_2(N)$ bits. After one observation of a challenge-response pair, if p is the probability that a randomly chosen passrule has the same PIN as s, then the entropy has been reduced to $\log_2(pN)$ bits. If $p \approx 0.0001$, this reduces the entropy by approximately $-\log_2(p) \approx 13.3$ bits. In [1], which considers the GrIDsure scheme, where no operators are employed, Bond points out that the maximal entropy $\log_2(N)$ is approximately 18.2 bits, so that, after two observations, it is likely that the adversary can determine the pattern fully. In [2], Currie points out that the inclusion of operators in the PassRules scheme, increases the maximum entropy so that the number of observations required to determine the passrule is also increased.

Currie outlines a simulation method, whereby given a passrule and a set of two challenge tables, one can obtain an estimate of the expectation of the number of passrules which are consistent with an observation of those. The logarithm of that number corresponds to the entropy associated with the adversaries uncertainty of the passrule after those observations. This assumes the adversary has no other information about the commonality of various passrules such as , for example, data on human factors related to the appeal or simplicity of certain passrules. This method can be extended to estimate this expectation for a given set of one passRule and n challenge tables, for any value of n.

The simulation method outlined is practical for very small values of n, where the fraction of consistent rules is relatively large, however, since this fraction decreases roughly as 10^{-rn}, where r is the number of available sub-rules from which to choose, this method is impractical for slightly larger n because the probability is so small. However, as n increases, the number of consistent rules approaches one very quickly, so that it makes sense to simply enumerate

the consistent rules rather than testing rules generated at random.

We apply simulation methods which are appropriate over the entire range of interest of values of n and we estimate the entire distribution of the number of consistent passRules as a function of the number of observation, instead of focusing strictly on the expectations of those distributions. The simulation method outlined in [2] estimates the expectation of the number of passrules consistent with a set of challenge tables, averaged over all possibilities for the true passrule. We will instead estimate how the number of consistent passRules varies with the number of observations for each choice of passRule individually. This identifies classes of passrules which are more or less secure with respect to a small number of observations and allow us to compare the entropy leakage profiles for different classes of passrules.

We also run simulations where we restrict the scheme to the top one hundred operator sequences (out of a total of 1,275 allowed sequences) in terms of entropy after three observations. We choose three observations because, after two observations, the entropy is usually still greater than that of a static four digit PIN, so that it is after the third observation that entropy leakage becomes a significant concern. We note that the top one hundred operator sequences include many minimum and maximum operators, which tend to yield the same output for a large number of different inputs. For this reason, we consider that, while we may improve the entropy associated with the uncertainty in guessing a passrule by restricting to the top one hundred rules, we may actually be doing worse with respect to the entropy associated with guessing the PIN. We therefore include simulation results on the entropy of guessing the PIN as well. Note that the problem of guessing the PIN, while related to that of guessing the passrule, is qualitatively different, in that an adversary who guesses a PIN may gain access once, while an adversary who guesses a passrule may gain access many times.

2 Simulation Method

We refer to one simulation of a series of observations of challenge-response pairs as a game. Each game corresponds to a simulation of one adversary attempting to guess the passrule of one user through these observations. We refer to the number of observations required for the adversary to determine the passrule as the length of the game.

The first step in each game is to choose a passrule. A passrule is chosen by first choosing a valid operator sequence out of the 1,275 operator sequences which satisfy the operator constraints, and then choosing the sequence of cells which the operators act on. The sequence of cells is chosen uniformly at random from those sequences with the appropriate number of cells and no cells reused. For reasons of efficiency, in order to minimize the memory required to run the simulations, we do not choose from operator sequences uniformly at random, however the distribution is approximately uniform and, in the analysis, we apply weights to the entropy distributions associated with different operator sequences to correct for this non-uniformity. In future work, we may need to alter the distribution by which operator sequences are chosen from uniform to a distribution which takes into account human factors such as preference for certain sequences. We leave this analysis for future work.

Each of the following steps in a game correspond to choosing a challenge table and determining the number of passrules which are consistent with the challenge-response pairs observed in the game so far. Challenge tables are chosen uniformly at random from the set of balanced tables by first choosing a table template from a list of 210 different options and then permuting the table template by a permutation of the 36 cells of the table, chosen uniformly at random. The table templates correspond to all of the different choices for the (ordered) contents of a balanced table, which correspond to all of the possible ways to choose the six digits which appear four times in the table as opposed to three.

There are 6,660 possible subrules, given the allowed operator set and the constraint that

cells cannot be reused. A passrule consists of four positions, each of which contains a subrule. At each observation in a game, we maintain, for each position, a list of the subset of the 6,660 subrules which are consistent with the sequence of PIN digits observed in that position in observations so far. Let N_i for $i = 1, 2, 3, 4$ denote the number of consistent subrules in position i after some number of observations. An upper bound on the number of passrules consistent with the observations so far can thus be given by

$$N = \Pi_{i=1}^{4} N_i$$

.

The actual number of consistent passrules is smaller than this number, because some of those subrule combinations either reuse cells or violate the operator sequence constraints. If N is relatively small, say on the order of 100,000 or so, we simply enumerate all of the possibilities to determine the number which satisfy the constraints, thereby determining the number of consistent passrules. If the number N is large, we sample from the population of size N to obtain an estimate $\hat{p}$ of the fraction p which satisfy the constraints and thereby estimate the number of consistent passrules as $N\hat{p}$. This is simply a binomial sampling exercise and we choose the sample size so that the relative error in estimation is within plus or minus one half of a percent of the actual value ninety nine times out of one hundred.

3 Simulation Results

We simulated seventy eight million games and recorded, for each game, the operator sequence and the number of consistent rules at each observation. The entropy at each observation in the game in bits is given by the logarithm base two of the number of consistent rules. Figure 1 illustrates the survival function for the entropy of a game after one, two, three and four observations. Figure 2 illustrates the survival functions of those distributions overlaid with

plots of normal distributions fit to the first two moments of those distributions. For three observations or less, the survival functions fit that of a normal distribution very closely and we see somewhat of a departure from normal for four observations. This departure from normal has mostly to do with the non-negative support of the entropy distribution and we leave for later examination the formulation of a simple model to fit this behavior.

Table 1 contains the mean and standard deviation of the entropy distributions for various number of observations. Table 2 contains the distribution of the number of observations required to determine a passrule completely. The most likely number of observations required is four, corresponding to nearly half of all cases, however, the distribution is somewhat skewed to the right. The mean number of observations required is 4.90 and the standard deviation is 1.50. Although it requires close to five observations on average to determine a passrule with certainty, after four observations, the possibilities will have been narrowed down significantly. The mean entropy after four observations is 1.21, which corresponds roughly to only 2 or 3 possibilities, which could often simply be attempted exhaustively in sequence by an adversary.

We also consider the distribution of the number of challenge-response pairs which can be observed without reducing the number of consistent rules to a number less than 10,000. This is of interest because the entropy is thus similar to that of choosing a four digit PIN uniformly at random. The mean number of these observations is 2.00 and the standard deviation is 0.113. The probability distribution for this quantity is given in Table 3. In almost 99 times out of 100, this number of observations is 2.

We also may be interested not just in the number of observations required for an adversary to determine a passrule, but also in the number of observations required to determine the PIN for the next challenge-response pair. For example, an adversary might observe three challenge-response pairs and narrow down the list of possible passrules to four or five, but upon receipt of the next challenge table, the adversary may notice that all of those passrules

resolve to the same PIN when applied to the table. In that case, the adversary would be able to gain one time access without observing any more responses. The distribution from the simulations of the number of challenge-response pair observations required to determine the next PIN uniquely in this manner is given in Table 9. The mean and standard deviation of this distribution are given by 4.71 and 1.09 respectively.

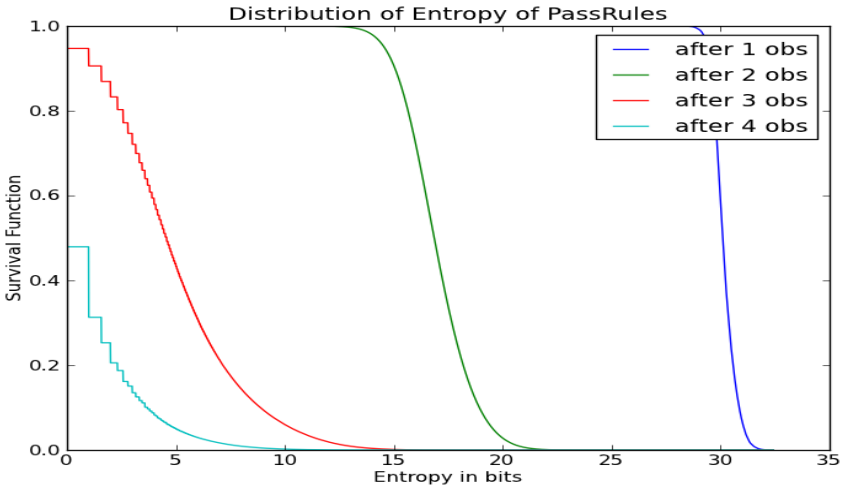

Figure 1: Distributions of Passrule Entropy

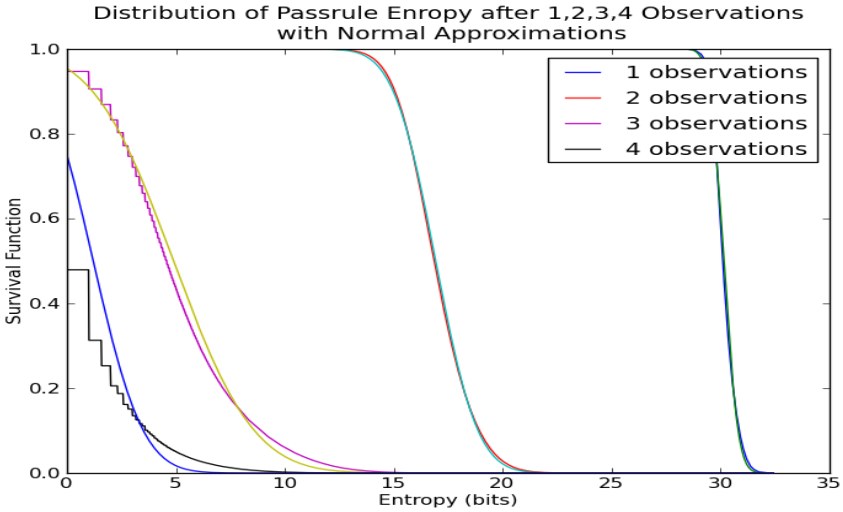

Figure 2: Distributions of Passrule Entropy with Normal Approximations

Table 1: Mean and SD of Entropy

Number of Observations	Mean Entropy (bits)	SD Entropy (bits)
1	30.16	0.51
2	16.94	1.52
3	4.90	2.90
4	1.21	1.80
5	0.46	1.06
6	0.20	0.65
7	0.094	0.41
8	0.047	0.27
9	0.026	0.19
10	0.019	0.16

Table 2: Distribution of Number of Observations to Determine Passrule

Number of Observations	3	4	5	6	7	8	9	10	>10
Probability	0.053	0.468	0.255	0.106	0.055	0.030	0.016	0.0085	0.0096

Table 3: Distribution of Number of Observations before Number of Consistent Rules Reaches 10,000

Number of Observations	1	2	3	4	5
Probability	0.0053	0.988	0.00716	6.04E-05	6.33E-07

4 Comparing Operator Sequences

We can also consider the entropy distribution as a function of the operator sequence. For the operator constraints currently considered, there are 1,275 valid operator sequences satisfying the constraints. For each of these operator sequences, we can estimate the distribution of entropy after n observations, for $n = 1, 2, 3, 4, \ldots$. Three observations seems to be a somewhat critical number in the sense that the entropy after two observations is rarely worse than a static four digit PIN and the number of possibilities after four observations has been narrowed down to only a very few. We thus consider, as a measure of the strength of an operator sequence, the mean entropy after three observations of games with rules corresponding to that operator sequence. Table 15 contains the operator sequence and the mean entropy after three observations for the top 40 operator sequences and for the bottom 40 operator sequences relative to this measure of strength. The symbols m,M,+,- and * represent the minimum, maximum, addition, subtraction and multiplication operators respectively and the digits 0-9 represent adding that digit to the contents of a cell modulo 10.

It is clear from comparing the top 40 and bottom 40 operator sequences that it is advantageous to have the minimum (m) and maximum (M) operators in a sequence. It is also clear that the unary operators, which use only one cell, are not particularly advantageous in that they are rare in the top 40 and very common in the bottom 40. It is somewhat surprising that there is an exception to this rule, in that the identity operator (0), which simply reproduces the contents of a cell is one of the most common operators in the top 40 sequences. This seems somewhat counter-intuitive, since it is the simplest operator in the list, however we conjecture that this has something to do with its similarity to the minimum and maximum operators, in that they also reproduce the contents of one cell or another without modification. We suspect that the advantage of the identity operator is realized

only because it can be confused, in a sense, with those by the adversary.

The fact that the simplest operator sequence, containing only the identity operator (0), is also the strongest (in the sense of mean entropy after 3 observations) could be a problem if it is known to users, as they may tend to favor choosing such a sequence, thereby skewing the distribution of operator sequences significantly from uniform. Note that the estimates for entropy all assume a uniform distribution of choice for operator sequences and will be affected by departures from this model. The human factors affecting the choice of operator sequences and rules is beyond the scope of this study.

It is tempting to add to the set of operator constraints to require the inclusion of advantageous operators such as the minimum and maximum operators, however such restrictions must be implemented with care, since it also decreases the number of valid rules overall and could therefore lead to an overall decrease in entropy. To investigate the trade-off between these two effects, we ran a simulation with approximately 2.5 million games, restricting the operator sequences to the top 100 sequences as defined above. We recorded the mean and standard deviation of entropy for each number of observations, and we compared these for the unrestricted case described above and the top 100 sequences only case. The results are recorded in Table 6.

The simulation restricted to the top 100 operator sequences out-performs the unrestricted case by approximately two bits of entropy at three observations. This increases the mean number of possible rules from on the order of 30 to on the order of 150 after three observations, which may well be significant, depending on the application. At four observations, the mean entropy is also increased by approximately two bits, so that the mean number of consistent rules increases from around two to around eleven on average. This may be significant in a case where, for example, three failed authentication attempts results in some alarm or action by the authenticator. At one or two observations, the mean entropy is slightly worse for the top 100 case, , by approximately six or two bits respectively. After one observation, this is

not very significant, since the entropy is quite high anyway. For two observations, it may be an issue, however the mean entropy still exceeds that of the static four digit PIN. The mean number of observations to determine a rule with certainty increases from just under five to 6.5, with a standard deviation of 1.6. On the other hand, the mean number of observations possible without decreasing the entropy below that of the four digit static PIN is decreased from 2.00 to 1.66. The distribution of the number of observations required to reach this stage is illustrated in Table 7, along with the same distribution for the unrestricted case. In the restricted case, the entropy drops below that of the static four digit PIN after two observations roughly one third of the time, whereas this is the case less than one percent of the time in the unrestricted case. The operator restrictions were intended to improve the entropy at the stage following three observations, however it may be advantageous to apply the same technique, but to attempt to optimize the case after two observations. This will likely decrease the probability of dropping below the 13.29 bit threshold (Entropy of static four digit PIN) after only two observations. We leave as future work further tuning of the set of allowed operators, however we have at least demonstrated that it is possible to improve certain aspects of the situation by that method and highlighted some of the trade-offs.

It seems worth-wile to investigate further into the restriction of operator sequences to some subset of possible sequences which are relatively stronger in some sense, such as in the sense explored here for example. We should note that we are concerned here with the entropy associated with the distribution of consistent passrules and so far have not considered the entropy associated with the distribution of PINs, the responses to the challenge tables. Consider a game where we have observed n challenge-response pairs and the $(n+1)^{st}$ challenge table but not the $(n+1)^{st}$ response (PIN). Since we know all of the consistent rules, we could, if we assume a uniform distribution for those rules, estimate the entropy associated with the distribution of PINS obtained by applying each possible consistent rule to the $(n+1)^{st}$ table. If p_i is the fraction of consistent rules which resolve to the PIN $i \in [0, 9999]$, the PIN or

response entropy is given by $E = \sum_{i=0}^{9999} p_i \log_2(p_i)$ bits. This entropy represents the entropy associated with guessing a PIN, as opposed to the entropy of guessing a rule, which we have considered so far. The relative importance of these two measures will depend in general on the nature of the application to which the passrules scheme is applied. We consider this distinction at this point because, while it seems that restricting to stronger rules improves the entropy of rules, it may actually decrease the entropy of PINS, because the nature of operators like minimum and maximum are that they tend to skew the distribution of digits lower or higher respectively.

Table 8 contains the means and the standard deviation of the distributions of PIN entropy for the unrestricted case and the restricted case where we have restricted to the top one hundred operator sequences as discussed above. For one or two observations, the mean entropy is decreased and the standard deviation is also increased, so that we are clearly doing somewhat worse for one or two observations. The mean in the restricted case is however larger and the standard deviations are comparable for three or more observations. After one observation, the number of PINs the adversary has to choose from is on the order of 6,000 in the unrestricted case and on the order of 1,000 in the restricted case. Probably, for most application, either of these will be acceptable. After two observations, this reduces to on the order of 1,800 or so in the unrestricted case and 100 or so in the restricted case. This difference between the unrestricted and restricted case may be significant enough to discourage the restriction. Again, this highlights the trade-offs, while some application dependent tuning of these parameters may be required.

The distribution of the mean number of challenge-resplonse pair observations required to determine the next PIN (given the next challenge table) is given in 9. The mean and standard deviation of the distribution are 5.93 and 1.09 respectively.

5 Multiple Clue Tables

A modification to the PassRules scheme has been suggested whereby the challenge is represented by a number of six by six tables, displayed all at once to the user in an array. Only one of these tables, corresponding to a particular position in the array known to the user is actually used to obtain the response PIN. In this scheme, the user applies the passrule to the table in the same position in the array at each authentication. We will refer to this position as the chosen position and alternatively to a table in that position as the chosen table. We will also refer to the other positions and tables as dummy positions and dummy tables. In this case, the adversary must determine not only the user's passrule, but also the chosen position in the array of tables. At each observation, the adversary must then determine a list of consistent passrules for each table in the array and the total number of passrules (or passrule-table-choice combination more correctly) is the sum of that number taken over the array of tables. Eventually, for dummy tables f, the number of consistent passrules reaches zero, while for the chosen table it converges to one. Before any observations, if we have an array of T tables, the entropy associated with guessing the chosen position and passrule is $\log_2(T)$ bits higher than that of the single table scheme. The number of tables which can be reasonably displayed simultaneously depends on the application but we might expect, for example $T = 9$, in which case we add on the order of three bits of entropy to the single table scheme at zero observations. After a few observations however, many of the position choices will have been eliminated, so that this entropy difference will be reduced. We ran simulations to estimate the rate of decrease of this added entropy in order to determine whether the multiple table scheme has any significant benefit.

We assume that each of the tables in an array are chosen independently from the others, so that the number of consistent rules for any table is also independent. We can therefore estimate the distribution of the total number of consistent rule-position-choice pairs by the

convolution of $T-1$ distributions for the number of consistent passrules for a dummy table and one distribution of the number of consistent passrules for a chosen table. We can estimate the distribution of the number of consistent passrules for a dummy table in the same way as that for the chosen table (described above) except that the sequence of PINs derives not from applying the passrule to the dummy table but to some other table, the chosen table. In each simulation, we obtain a sequence which converges to zero instead of to one, as is the case for a chosen table.

We simulated 75,000 games with a dummy table. Figure 3 illustrates the conditional distribution of the entropy associated with the number of consistent rules after 1,2,3 or 4 observations, conditional on that number being greater than zero (since otherwise the entropy is undefined). Table 4 contains the mean and standard deviations of those conditional distributions. The maximum number of observations from our simulations required to eliminate the dummy table was five and the mean and standard deviation of that number were 3.45 and 0.50. The distribution of the number of observations required to eliminate a dummy table is given in Table 5. The number of observations required to eliminate a dummy table is split roughly in half between three and four observations.

Figure 4 illustrates the distribution of the passrule entropy for the system with nine tables: one chosen table and eight dummy tables. The distributions are similar to those without the dummy tables, except that they are shifted two or three bits to the right and the distribution after four observations appears more Gaussian than it did without the dummy tables. The means and standard deviations of those distributions are given in Table 10. The addition of the dummy tables improves the entropy by about three bits after one or two observations, although this is not particularly significant, since the entropy is still relatively high at that point, roughly equivalent to a seven digit static PIN on average after two observations. The more significant effect of the dummy tables is the increase in the mean by approximately 2.25 bits after three observations . This increases the number of consistent rules which the

Table 4: Conditional Moments of Entropy for Dummy Table

Number of Observations	Mean Entropy (bits)	SD Entropy (bits)
1	29.94	0.53
2	16.22	1.41
3	3.42	2,07
4	0.99	1.21

adversary must choose from to guess the passrule after three observations from, on average, approximately 30 to on the order of 140 or so. In addition, the standard deviation is reduced after three observations with the dummy tables, so this will reduce the number of times that the number of consistent rules falls much below 30. The form of the entire distributions with and without the dummy tables should be compared in the context of the particular application of interest to determine if the addition of the dummy tables is worth-wile or not for a particular case. After four observations, the result is almost identical with or without the dummy tables, since all dummy tables will have been identified by that point almost 99 percent of the time.

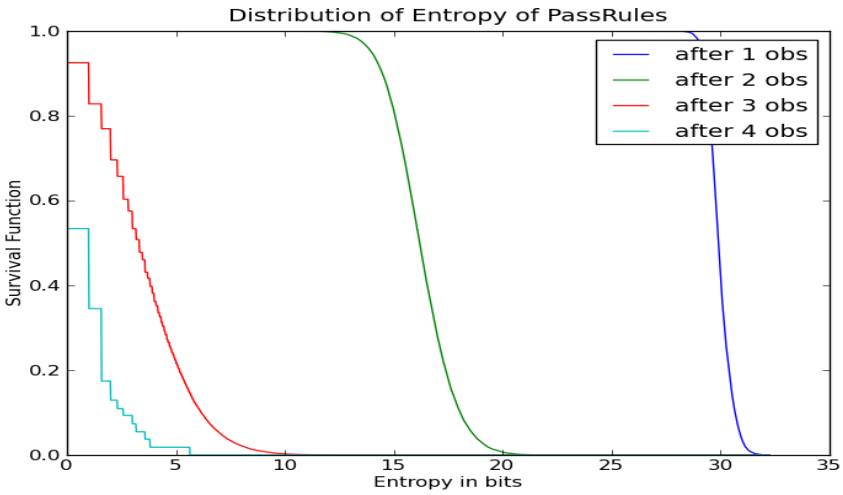

Figure 3: Distributions of Conditional Passrule Entropy For a Dummy Table

Table 5: Distribution of Number of Observations to Eliminate Dummy Table

Number of Observations	3	4	5
Probability	0.554935342	0.44441148	0.000653178

Table 6: Passrule Entropy Comparison with Top 100 Operator Sequence Restriction

Number of Observations	All Sequences Mean Entropy (bits)	SD Entropy (bits)	Top 100 Sequences Mean Entropy (bits)	SD Entropy (bits)
1	30.16	0.51	24.13	1.53
2	16.94	1.52	14.11	2.43
3	4.9	2.9	7.26	2.63
4	1.21	1.8	3.46	2.17
5	0.46	1.06	1.6	1.56
6	0.2	0.65	0.72	1.05
7	0.09	0.41	0.32	0.68
8	0.05	0.27	0.14	0.44
9	0.03	0.19	0.06	0.28
10	0.02	0.16	0.03	0.18

Table 7: Distribution of Number of Observations before Number of Consistent Rules Reaches 10,000

Number of Observations	1	2	3	4	5
Unrestricted case Probability	0.0053	0.988	0.00716	6.04E-05	6.33E-07
Restricted case Probability	0.356	0.630	0.014	1.47E-04	1.21E-06

Table 8: PIN Entropy Comparison with Top 100 Operator Sequence Restriction

Number of Observations	All Sequences Mean Entropy (bits)	SD Entropy (bits)	Top 100 Sequences Mean Entropy (bits)	SD Entropy (bits)
1	12.64	0.23	9.99	0.89
2	10.82	0.71	6.85	1.61
3	3.57	1.87	3.79	1.71
4	0.75	1.12	1.86	1.41
5	0.26	0.66	0.88	1.02
6	0.11	0.41	0.4	0.7
7	0.05	0.26	0.18	0.47
8	0.03	0.18	0.08	0.31
9	0.02	0.14	0.03	0.2
10	0.01	0.11	0.01	0.13

Table 9: Distribution of Number of Observations to PIN Certainty

3	4	5	6	7	8	9
0.05	0.47	0.29	0.11	0.05	0.02	0.01

Table 10: Distribution of Passrule Entropy with 8 Dummy Tables

Number of Observations	1 Table Mean Entropy (bits)	SD Entropy (bits)	Nine Tables Mean Entropy (bits)	SD Entropy (bits)
1	30.16	0.51	33.24	0.19
2	16.94	1.52	20.11	0.63
3	4.9	2.9	7.16	1.93
4.	1.21	1.8	1.21	1.80

Table 11: Summary of Game Lengths

Simulation	Number of Observations to					
	Guess Passrule		Guess PIN		$> 10,000$ PINs	
	Mean	SD	Mean	SD	Mean	SD
Base Model	4.9	1.5	4.71	1.09	2	0.11
8 Dummy Tables	4.96	1.44			2.01	0.18
Top 100 Operator Sequences	6.5	1.6	5.93	1.09	1.66	0.5

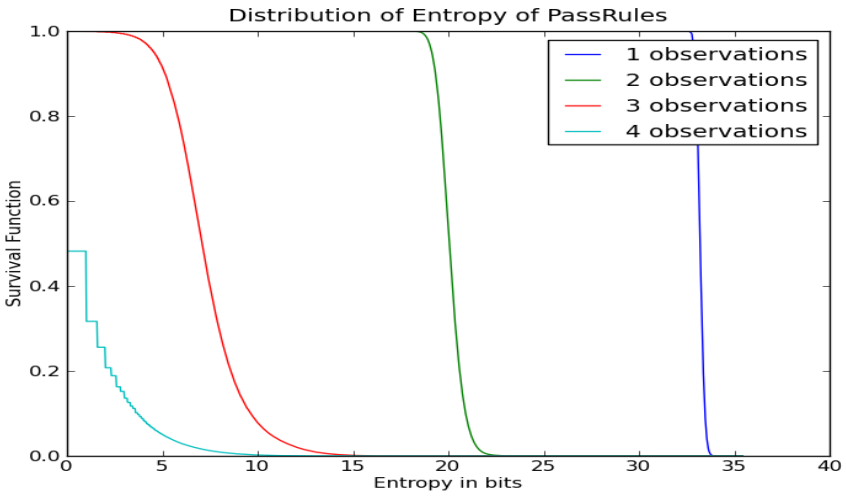

Figure 4: Distributions of Passrule Entropy with 8 Dummy Tables

6 Omission of Operator Sequence Constraints

The current version of the Passrules scheme includes some constraints on the form of operator sequences which can be used for passrules. Passrules cannot reuse cells, passrules must use at most two distinct operators and passrules must have at most two transitions between operators. Intuitively, if we omit these restrictions, the entropy associated with guessing a passrule will increase and the number of observations required to guess a rule will also increase (in a stochastic sense).

Table 12 contains the mean and standard deviation of the entropy associated with guessing a passrule after a number of observations. When compared to the scenario where the constraints are enforced, we see that the constraints decrease the mean entropy by on the order of 4 bits.

Figure 5 illustrates the distributions of entropy after a number of observations.

Table 13 contains the distribution of the number of observations required to guess a passrule. The mean and standard deviation of this number are 5.92 and 1.87 respectively. Thus the operator constraints decrease the number of observations required by on the order

Table 12: Mean and SD of Entropy with no Operator Sequence Restrictions

Number of Observations	Mean Entropy (bits)	SD Entropy (bits)
1.0	33.8	0.45
2.0	20.67	1.27
3.0	8.7	2.23
4.0	2.43	1.9
5.0	0.89	1.27
6.0	0.44	0.88

Table 13: Distribution of Number of Observations to Determine Passrule without Operator Sequence Constraints

Number of Observations	3	4	5	6	7	8	9	10	>10
Probability	2.7e-05	0.154	0.409	0.184	0.096	0.061	0.039	0.024	0.033

of 1 on average. Similarly, Table 14 contains the number of observation which can occur without decreasing the entropy below that of a 4-digit PIN. The mean and standard deviation for that number are 2.02 and 0.15 respectively, so that the constraints have very little effect on this measure on average.

7 Phishing

Phishing has been identified in [2] as a significant threat to the security of passrules. A number of schemes were identified which could be used to identify a passrule in a case where an adversary can provide a tailored grid and obtain the corresponding PINs. Schemes were identified which could identify a passRule with as few as three challenge-response pairs with grids tailored by the adversary. This threat can be ameliorated with the use of a variation

Table 14: Distribution of Number of Observations before Number of Consistent Rules Reaches 10,000 without Operator Sequence Constraints

Number of Observations	1	2	3	4	5	>5
Probability	1.46e-07	0.977	0.023	2.83e-05	2.2e-07	0.0

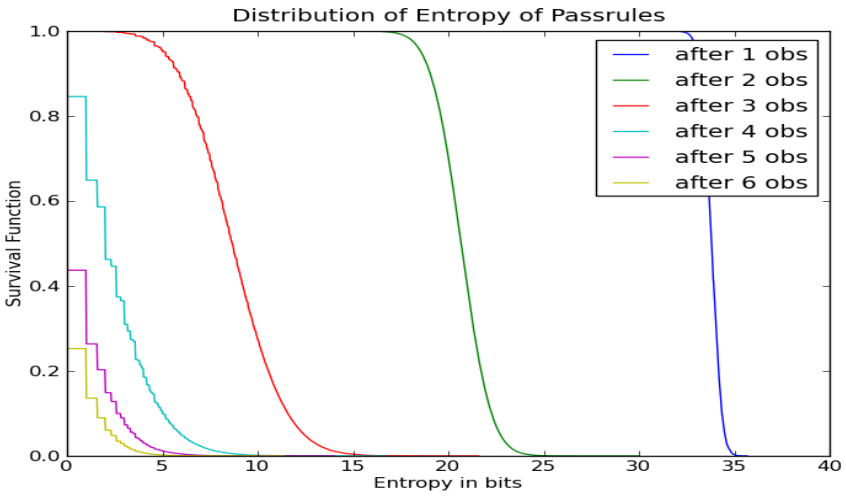

Figure 5: Distributions of Passrule Entropy with no Operator Constraints

of a checksum on either the challenge table or the PIN.

One option is to define a second passrule and a fixed check PIN. Since challenge tables are pre-calculated, it would be a simple matter to use only tables for which the second passrule resolves to the check PIN. If the user suspects there may be a phishing attempt in progress, he can simply evaluate the second passrule and check that it matches the check PIN. An adversary, since he never observes the check PIN, would have to receive two responses to gain any information on the the second passrule or check PIN. The probability of the adversary choosing two valid tables in sequence would be on the order of one in a hundred million, which is quite small. The application of a second passrule may be considered onerous by some users, however it would useful in a case where the user may not apply the check every time he authenticates but only when something about the authentication process seems to be different, or when he already has some suspicion, for whatever reason. In ths case, he will at least have a method whereby he can check whether or not he is being phished.

It is also possible to use the same idea, but to apply the check rule not to the entire table but to the PIN itself. For example, we could apply a two-subrule passrule to digits chosen from the four digit PIN. This would be significantly easier to remember and less onerous to

execute. If we don't disallow reuse of cells for this, there are then 70 possibilities for each of the two subrules and therefore 4,900 possible check rules of length two. There is a one in 10,000 chance that an adversary could randomly choose two valid tables in a row without any a-priori knowledge. If he were to do this, he would be able to narrow down the list of possible check rules to on the order of 49, since only rules which resolve to the same value for both tables need be considered. His chances of getting three valid tables in a row is therefore on the order of one in 490,000. This is probably sufficient for many cases, provided the user is checking the validity of the tables. If the user does not check the first few tables, the argument proceeds starting from the first table that he checks, so that the probability of getting three responses from the user once he starts checking is on the order of one in 500,000. This is improved somewhat by the fact that the adversary does not know which of his tables are valid, because he does not know when the user was checking or not.

There are numerous possible variations of this scheme and, for the most part we expect the security to increase as a function of the complexity of the check rule that is applied. Such a scheme would have to be tailored to the application and could in fact be tailored to individual users in some cases. In any case, the user will incur some cost of complexity or annoyance in order to implement the scheme.

8 Conclusion

We have quantified through simulation the entropy leakage associated with observations by an adversary of challenge-response pairs of a passrule authentication scheme. We have also investigated the variation of this entropy leakage with the form of the passrule chosen and investigated a scheme which restricts passrules to those which are relatively less affected by entropy leakage. We considered the entropy associated with an adversary's attempt to guess a passrule as well as that associated with an attempt to guess the response to a

given challenge table and we consider a scheme with multiple dummy challenge tables. The schemes associated with dummy tables and with restricting the allowed passrules both seem to have some effect on the rate of entropy leakage, but more work needs to be done to tune a system to take advantage of those concepts. A scheme to combat phishing attempts was also described.

Table 11 contains a summary of the mean and standard deviation of the number of observations required to guess a passrule or a PIN and of the number of observations that can be observed without decreasing the number of consistent passrules to less than 10,000, which is the number of possibilities for a static four digit PIN. The table contains these measures from the simulations for the base case as well as for the case with multiple tables and the case where operator sequences are restricted to the top 100 of those. Those values for the number of observations required to guess a PIN in the multiple tables case are not included, since we did not construct a simulation for that case. In that case, the entropy for the system is a function not just of the sum of the number of possibilities for each case, but of the distribution of PINS in the entire collection of tables and so we need to explicitly simulate each of the multiple tables and cannot make use of the independence of tables. This simulation is certainly possible, but outside the scope of this particular excercise.

One of the challenges in attempting to decrease the rate of entropy leakage stems from the independence of subrules in a passrule. An adversary can collect information and narrow down the set of possibilities for each subrule independently, so that the number of observations required to guess a passrule is simply the maximum of the numbers of observations required for each separate rule. If we were to break down this independence, by constructing passrules where each subrule depends somehow on the result of evaluating the previous subrule in the sequence, we may gain some ground in reducing the rate of entropy leakage. In that case, an adversary would need to maintain and search some sort of tree structure, in order to keep track of the list of consistent passrules. We leave the exploration of this

conjecture for future work.

Table 15: Mean Entropy after 3 Observations for Top and Bottom 40 Operator Sequences

Top 40					Bottom 40				
S1	S2	S3	S4	Mean Entropy	S1	S2	S3	S4	Mean Entropy
0	0	0	0	12.79	5	5	4	4	2.53
0	0	m	0	11.84	4	4	2	2	2.53
0	m	0	0	11.84	5	6	6	5	2.53
0	0	0	m	11.83	8	5	5	8	2.53
m	0	0	0	11.82	5	4	4	5	2.53
M	0	0	0	11.73	4	4	5	5	2.53
0	0	0	M	11.73	4	6	6	4	2.53
0	0	M	0	11.72	4	4	6	6	2.54
0	M	0	0	11.7	3	3	5	5	2.54
m	0	0	m	11.01	6	5	5	6	2.54
m	m	0	0	11.01	8	4	4	8	2.54
0	0	m	m	11.01	6	6	5	5	2.54
0	m	m	0	11.0	6	6	4	4	2.54
M	0	0	M	10.84	3	5	5	3	2.54
0	0	M	M	10.82	4	5	5	4	2.54
0	M	M	0	10.8	6	2	2	6	2.54
M	M	0	0	10.8	7	7	6	6	2.54
m	m	0	m	10.45	6	6	8	8	2.54
m	0	m	m	10.44	4	1	1	4	2.54
m	m	m	0	10.42	5	5	6	6	2.54
0	m	m	m	10.41	2	2	3	3	2.54
m	m	m	m	10.12	5	5	3	3	2.54
M	M	0	M	10.04	3	4	4	3	2.55
0	0	0	*	10.04	2	5	5	2	2.55
0	M	M	M	10.04	2	3	3	2	2.55
0	0	*	0	10.03	3	3	4	4	2.55
0	*	0	0	10.03	7	7	5	5	2.55
M	M	M	0	10.03	5	5	7	7	2.55
*	0	0	0	10.03	3	3	6	6	2.55
M	0	M	M	10.01	3	3	2	2	2.55
0	-	0	0	9.59	6	8	8	6	2.55
0	0	0	-	9.58	2	2	4	4	2.55
-	0	0	0	9.57	8	8	5	5	2.55
0	0	-	0	9.55	1	1	2	2	2.55
M	M	M	M	9.34	6	4	4	6	2.55
+	0	0	0	9.31	6	6	1	1	2.55
0	0	0	+	9.28	8	6	6	8	2.55
0	0	+	0	9.28	2	4	4	2	2.55
0	+	0	0	9.28	3	2	2	3	2.55
0	0	4	0	9.11	8	8	2	2	2.55
0	0	7	0	9.11	8	2	2	8	2.55

References

[1] M. Bond. Comments on gridsure authentication. `www.cl.cam.ac.uk/~mkb23/research/GridsureComments.pdf`, March 2008.

[2] James D. Currie. Security of passrules. *Internal Report for PassRules Canada Inc.*

[3] R. Weber. The statistical security of gridsure. `www.gridsure.com/resources/analyst/analystreport-weber.pdf`, June 2006.