Why didn't my (great!) protocol get adopted?*

Mehdi Nikkhah
ESE, University of Pennsylvania
mnikkhah@seas.upenn.edu

Constantine Dovrolis
CS, Georgia Institute of Technology
constantine@gatech.edu

Roch Guérin
CSE, Washington University in St. Louis
guerin@wustl.edu

ABSTRACT

What determines the eventual success of a protocol? Are certain features or properties more important? Do those vary according to a protocol's type? We explore these questions by applying data mining techniques to a rich repository of protocol specifications; IETF RFCs. While the investigation is still preliminary, some interesting findings have emerged. It confirms a number of intuitive results such as backward compatibility being key for protocol extensions and new versions, but not for new protocols. Similarly, the ability to improve performance is the single most important factor in the success of data plane protocols. Less intuitive findings, however, also emerge. Adding value to other protocols was the most significant factor in the success of new protocols, while extensions targeting security were the most likely to fail among new application and transport layer protocols. The paper offers a brief overview of our methodology and of the initial results it has afforded.

Categories and Subject Descriptors

C.2.6 [**Computer-Communication Networks**]: Internetworking—*standards*

General Terms

Standardization, Design

Keywords

Protocols, adoption, machine learning

*The work of the first and third authors was supported by NSF under grant CNS-1361771, while the work of the second author was supported by NSF under grant CNS-1319549 and by a gift from Cisco Systems.

HotNets '15 November 16–17 2015, Philadelphia, PA USA

DOI: http://dx.doi.org/10.1145/2834050.2834103

1. INTRODUCTION

Over the past decades, the networking community has learned much about protocol design. However, we know much less about what controls a protocol's success in the "real world". IPv6 is a well-known instance, which more than two decades after its introduction still struggles to achieve wide adoption. And there are many other examples. Since 1969 the Internet Engineering Task Force (IETF) has produced over 3100 *standards track* Request for Comments (RFCs). However, in spite of a rigorous vetting process, success, *i.e.,* wide adoption by their target audience, has eluded close to half of them[1].

This raises important questions that, except for the investigation of RFC 5218 [10], have received little attention to-date[2]. In particular, are specific features or properties more important than others when it comes to influencing a protocol's success? Clearly, technical correctness is important, but we have arguably made much progress in weeding out flawed protocols. External factors such as luck or commercial interests will always be present, but are unlikely to translate into systematic biases. The question is whether it is possible to carry out a quantitative and statistically rigorous investigation of protocols and protocol extensions[3] to identify factors with a significant influence on their success (or failure).

In this paper, we apply statistical tools to mine a rich and diverse repository of protocols, namely *standards track* RFCs. Standards track RFCs correspond to protocols that have progressed through rounds of discussions in an IETF Working Group (WG), and been deemed stable and significant enough to warrant formal publication. This should, therefore, eliminate technically flawed protocols, as well as those with little community support. Our goal is to identify statistically significant features that play an important role in a protocol's success, with success defined as "broad-based"

[1]A random sample of close to 200 standard tracks RFCs yields a success rate of about 60%.
[2]The other related body of works is "network economics", *e.g.,* [6], that has a very different focus.
[3]For conciseness and unless otherwise warranted, we use the term protocol to refer to both *new* protocols and extensions or new versions of *existing* protocols.

adoption among its intended users. Note that in identifying such features, we do not seek to build a prediction tool. Instead, we aim to offer guidance to protocol designers by highlighting features that may be of particular significance for different types of protocols.

Our approach is three-prong. We first identify features that reflect protocol characteristics, which *may* play a role in their success. Crafting such a list is a somewhat subjective process that borrows on our experience with protocols and protocol design. We discuss our approach and its outcome in Section 2. Next, we construct a data set for our statistical analysis. This data set is built from a sample of standards track RFCs (Section 3 discusses the selection process), which are then characterized in terms of their features and labeled as successful or not. Finally, as described in Section 4, we apply a well-established classification framework to extract protocol features that show statistically significant correlation with the success or failure of protocols. The results are analyzed in Section 5 to explore their implications, as well as perform limited validation.

2. PROTOCOL ADOPTION FEATURES

Network protocols span many functionalities, designs, and implementations. How do we capture features that set them apart, possibly influencing adoption outcomes? In this section, we put forward a nomenclature that incorporates both traditional differentiators used when describing protocols, *e.g.,* the layer they target, and aspects of value and dependencies on other protocols.

2.1 Characterizing Protocols

We characterize features that may affect a protocol's adoption along three axes: **(i)** functionality and role; **(ii)** impact and/or dependency on other protocols; and **(iii)** value and how it is realized. A protocol's functionality and role affect when and where it is needed. How and how much a protocol interfaces to other protocols or requires them to change can make adoption harder or easier. Finally, the benefits that a protocol affords its users and when it allows them to accrue those benefits likely also plays a major role in its adoption.

The first axis partitions protocols according to the position (layer) they occupy in the protocol ecosystem. The second characterizes a protocol based on its interactions with other protocols, including earlier versions of itself, when applicable. The third axis reflects the protocol's functionality and its ability to realize its value.

Before detailing the resulting list of features, we first highlight two key properties we enforce to facilitate statistical analysis. Specifically, protocol features must be: **(i)** *Binary*[4]: This is to lower the "noise" inherent when measuring continuous valued functions.

[4]Categorical features (*e.g.,* protocol type) can be transformed to binary features.

(ii) *Objectively measurable*: This is again intended to limit the measurement noise that arises when subjective assessments are used to set a feature's value.

2.2 Features List

In investigating factors that may be influencing adoption, we consider the following twenty protocol features:

2.2.1 Protocol Functionality

The concept of layering has played a strong role in the design of communication protocols, and accordingly layers provide rough boundaries along which to partition protocols and their functionalities. Additional categories are, however, necessary to distinguish between protocols with similar functionality but targeting different users, *e.g.,* end-users vs. the network itself. This led to the formulation of the following six features or categories under which to classify protocols:

(1) Application (A): Protocols that provide different means of network communication to users, *e.g.,* ssh;

(2) Transport (T): Protocols that deal with end-to-end connectivity functionality, *e.g.,* TCP;

(3) Network Services (S): Protocols that target services that facilitate network use, *e.g.,* DNS;

(4) Network Control Plane (C): Protocols that affect network configuration, *e.g.,* RSVP;

(5) Network Routing Plane (R): Protocols that determine packet forwarding, *e.g.,* BGP;

(6) Network Data Plane (D): Protocols that deal with packet format and handling, *e.g.,* IPv6

This classification omits Link and Physical layer protocols. The primary reason is that the IETF targets very few protocols in these categories, with IEEE and other bodies the main venues for their standardization.

2.2.2 Impact or Dependency on Other Protocols

These features capture the impact or dependency of a protocol on other protocols or network components, as well as its interactions with them. The information needed to identify them can be extracted from the protocol's RFC and accompanying documents.

(7) New protocol vs. extension/version of an existing protocol: A new protocol is a clean design, while an extension/version of a protocol inherits and/or builds on its predecessor. This creates different adoption challenges, *e.g.,* extensions/versions need to interact with an installed-base, while new protocols may need to displace a functionally similar protocol.

(8) Replacing another protocol: This applies to new and existing protocols seeking to replace an existing one. Displacing an incumbent can be challenging.

(9) Requiring changes to other protocols: A protocol may require changes to other protocols, *e.g.,* TCP's Explicit Congestion Notification affects the IP header, which creates additional adoption hurdles.

(10) Generating value for other protocols: A protocol can benefit others, *e.g.,* IPSEC offers security to upper layer protocols, which can help adoption.

(11) Affecting network (hardware or software) components: A protocol may require changes to network components, *e.g.,* IPv6 impacts router hardware and software. Deploying them can delay adoption.

(12) Backward compatibility: This applies to protocol extensions/versions and their ability to interoperate with earlier versions, *e.g.,* TCP SACK. The extent to which they can likely affects adoption.

2.2.3 Value and its realization

We distinguish between three main value categories, namely, performance, security, and scalability, plus one "catch-all" category.

(13) Performance: This covers improvements to communication throughput or latency, *e.g.,* as in some TCP extensions.

(14) Security: This includes authentication and encryption aspects, as well as protocol mechanisms aimed at strengthening communication integrity.

(15) Scalability: Accounts for a protocol's ability to efficiently operate at Internet's scale.

(16) Others: A place-holder for motivations that do not belong to any of the above three categories.

While the above features identify where the value of a protocol might lie, another important aspect is realizing this value. Does value grow with adoption, and if yes, how? We distinguish between four different scenarios.

(17) Local: Full value is realized even under limited (individual) adoption. Mobile IP can, for example, be argued to fall in this category.

(18) Domain-wide: Adoption within the realm of a single management entity, *e.g.,* an Autonomous System, is sufficient to unlock the protocol's value. This is common with intra-domain routing protocols.

(19) Internet-wide: Realizing the bulk of the protocol's value calls for widespread adoption, *e.g.,* as is the case with IPv6.

(20) Increasing: The value of many protocols increases with adoption, *e.g.,* the benefits of DNSSEC grow, the more widely adopted it is.

3. DATA COLLECTION

We construct a representative sample of RFCs in three steps. The first eliminates all RFCs issued after 2009. This ensures enough time has elapsed since the protocol's initial release to reasonably assess its adoption status. The second step focuses on producing an unbiased subset of RFCs. This relied on combining two sampling methods, each producing 100 RFCs.

The motivation for using two different sampling methods is to avoid over-sampling "popular" protocols that tend to see many extensions and correspondingly generate a large number of RFCs, *i.e.,* a random sampling tends to sample popular protocols more often than other protocols, thereby introducing a bias that favors factors correlated with those protocols. The first sampling method (referred to as "random") randomly samples all pre-2009 standards track RFCs. The second method (referred to as "WG-based), first selects one major[5] RFC from each (active or archived) IETF WG, and then randomly samples the resulting set.

The third step is a data sanitization step applied to both samples. It removes duplicate instances of RFCs and "irrelevant" RFCs, *e.g.,* RFCs offering ancillary information on a protocol such as Management Information Bases (MIBs). RFCs dealing with Link or Physical layer protocols were also removed in this step. The final set of RFCs included 173 distinct RFCs.

The next and most time consuming step consists of characterizing the protocol of each RFC according to the features of Section 2, and label it as successful or not. This is a manual step that relies on our own experience with protocols, and a broad range of external sources, *e.g.,* books, technology blogs, source code forums, product web pages, IETF mailing lists, etc. This process, while lengthy, was to some extent simplified by the fact that we dealt primarily with binary decisions for each feature. This mitigated the impact of unavoidable inaccuracy in the information that led to classifying each protocol as either having or not having a particular feature. The result of this classification is available in the form of a spreadsheet[6]. Each row in the spreadsheet maps to one of the 173 protocols under consideration, while columns correspond to the features of Section 2 and to the label of successful vs. not successful. Many cells include comments highlighting the motivations behind their setting and/or pointers to documentation supporting the choice that was made.

As protocol functionality arguably represents a natural partitioning, Table 1 presents classification results for functionality features, together with the corresponding number of protocols deemed successful in our two samples, random and WG-based.

RFCs	A	T	S	C	R	D	Succ.
Random (83)	24	8	21	3	22	5	55
WG-based (90)	26	6	24	11	7	16	52

Table 1: Classification statistics by functionality

4. METHODOLOGY

Given our relatively small dataset and our goal of identifying features that play a major role in a proto-

[5]Major RFCs are either a new protocol or a significant change to a protocol, independent of their success.

[6]`https://docs.google.com/spreadsheets/d/1JZzryCZ0h52iPgHqOFcn2HHAUpCC5dyUnf2qrAJc8oE`.

col's adoption, we considered statistical methods such as logistic regression, decision trees, or Logistic Model Trees (LMT), instead of less-transparent and more data demanding algorithms such as neural networks and K-nearest neighbor [11]. We eventually settled on binary logistic regression[7], since our focus is on the interpretability of the results (*i.e.,* regression is more transparent in terms of which factors/features are significant, and in quantifying their relative effects).

The relatively large number of features we rely on to characterize protocols[8] (see Section 2), leads us to first use stepwise regression to isolate features (a model) with the highest classification impact. Stepwise regression adds features one-by-one in its forward mode and removes them, also one-by-one, in its backward mode, and at each step examines whether a particular criterion, *e.g.,* Akaike Information Criterion (AIC), or Bayesian Information Criterion (BIC) [7], is minimized[9]. In our analysis, we focus on AIC (essentially a measure of the model's quality based on a trade-off between its goodness of fit and its complexity), as it proved the most efficient in selecting a model. Results for BIC and other criteria were, however, qualitatively similar.

Once key features have been identified, we feed them to the binary logistic regression. A positive (negative) outcome corresponds to an RFC classified as successful (unsuccessful). We rely on JMP 12 [2] for stepwise regression and binary logistic regression, and for the statistics it provides to characterize the outcome of the classification. Weka 3-7-12 [3] is used for cross-validation (see below), and to generate a confusion matrix. Because of our relatively small data set (and large number of features), we can face a quasi-separation problem where the model overfits the data. It memorizes the data instead of learning the relationship between the response and the features. The resulting model coefficients are then not statistically significant, and the model offers little classification value. When faced with such a scenario, we rely on built-in regularization tools in Weka and JMP, namely, Ridge Regression and Firth Bias-Adjusted Regression, to avoid overfitting.

The outcomes identify which features play important roles in a protocol's success or failure through two main metrics, odds ratio and statistical significance (*i.e.,* p-value). The odds ratio captures the odds that an outcome occurs given the presence of a particular feature, compared to the odds of the outcome occurring in the absence of that feature; odds ratio values less (greater) than 1 imply the existence of a negative (positive) correlation. The statistical significance of each feature is characterized through a likelihood ratio test. This test compares the model's likelihood to that of an alternative model from which the feature is absent. The p-value is then obtained assuming a χ^2 distribution for the test statistic. The smaller the p-value, the less likely the alternative model from which the feature is absent[10].

The odds ratio and p-value reflect our focus on identifying features likely to play an important role in a protocol's success, rather than developing a "predictor" for a protocol's eventual success. However, we also consider metrics that evaluate the model's predictive accuracy.

The first is the (random) 5-fold cross-validation accuracy [8]. 5-fold cross-validation randomly divides the dataset into 5 equal subsets, and uses each subset as a test set, with the other four used to train the classifier. The classification rates on each test set are averaged to build the 5-fold cross-validation classification accuracy.

Another relevant metric is the "confusion matrix" that consists of True Positive (TP), False Negative (FN), True Negative (TN), and False Positive (FP) rates. It measures the classifier's ability to correctly identify successful and unsuccessful protocols. TP (TN) is the fraction of successful (unsuccessful) RFCs properly classified. Conversely, FP (FN) is the fraction of unsuccessful (successful) RFCs classified as successful (unsuccessful).

5. INITIAL RESULTS

This section reports results from applying the classification of Section 4 to the 173 RFCs in our data set.

The (random) sampling process that generated our 173 RFCs, arguably resulted in a disparate set of protocols. This reflects protocol diversity, but makes it unlikely that the same features are behind the success (or failure) of each one of those 173 protocols. This begs the question of whether seeking to identify a common set of features is meaningful in the first place. And if not, how should we instead group protocols?

We explore this issue by first applying our classifier to the full set of 173 protocols, and then separately to new and extensions or new versions of existing protocols. Stepwise regression generated a relatively large set (6 and 8) of "relevant" features when applied to either all or to only existing protocols, but somewhat surprisingly only one feature for new protocols. The large number of features for the first two categories is consistent with our expectations given the underlying protocol diversity, and makes interpreting the results difficult. For new protocols, the one feature singled-out was *"adding value to other protocols,"* which, as we shall see next, emerges as a common theme for many sub-categories of protocols. Hence, pointing to its likely importance across new protocols of different types.

[7]"The binary logistic model is used to predict a binary response based on one or more predictor variables (features), making it a probabilistic classification model in the parlance of machine learning" [1].

[8]As a pre-analysis step, we adjusted the set of features to remove any multi-collinearity between them.

[9]We also considered p-value thresholds, which enters (removes) a feature to (from) the model only if its significance meets the "enter" or "leave" thresholds.

[10]We target a significance $\geq 95\%$, *i.e.,* a p-value ≤ 0.05.

Odds ratios were middling (mostly in the 2-4 range for positive correlations, and similarly for negative correlations), but improved slightly when separating protocols into new and existing versions. A similar pattern was observed for p-values. A few values fell below the target 95% confidence, and separating protocols into new and existing again only produced minor improvements.

This motivated grouping protocols into more consistent sets, whose success would then more likely depend on similar features. A natural grouping is along a protocol's functionality, or closely aligned with it[11]. We rely on such a grouping, and report next the results of our classification for each group. Results for the first two groups are split between new and existing protocols, but aggregated for the last three groups primarily because they include too few protocols (something we plan to address in the future). The results are presented in the form of tables with rows listing the features identified by stepwise regression, and for each feature highlighting its odds ratio (OR) and statistical significance (p-value). The last two rows report the 5-fold cross validation accuracy (Accuracy) and the confusion matrix.

5.1 Application & Transport Layer Protocols

Our first group combines application and transport (A & T) layer protocols. They share many properties, *e.g.,* both reside primarily in end-systems, but more particularly, have in common that their value usually increases with adoption. This is rarely so with other types of protocols.

Feature	OR	p-value
Value for other prot.	12.60	0.014
Accuracy	72.7%	
Confusion Matrix	TP= 0.58 FN= 0.42 TN=0.90 FP= 0.10	

Table 2: New A & T protocols (22 RFCs)

Table 2 reports results for new protocols and highlights that adding value to other protocols is the single most important factor behind the success of a new A & T protocol. In hindsight, this may be intuitive for transport protocols which need to demonstrate value to applications (and application protocols) to be adopted, *e.g.,* RTP adds value to SIP and other streaming application layer protocols. This is less so for application layer protocols, though many interact with other application layer protocols; in the process potentially contributing value to those protocols. Note also that the importance of this feature does not mean it is mandatory for a protocol's success. As a matter of fact, the high FN value indicates that close to 40% of successful protocols did not have it.

[11]As a sanity check, "arbitrary" groupings were also investigated, and consistently yielded poorer outcomes.

Features	OR	p-value
Backwards compatibility	48.51	0.001
Security motivation	0.046	0.011
Accuracy	83.3%	
Confusion Matrix	TP= 0.88 FN= 0.12 TN=0.63 FP= 0.37	

Table 3: Existing A & T protocols (42 RFCs)

Results for existing A & T protocols are presented in Table 3. They show that backward compatibility and security-motivated extensions influence success positively and negatively, respectively. Both are relatively intuitive. We expect backward compatibility to be important for existing protocols. Conversely, we know from experience the struggles that security-motivated protocols commonly face, *e.g.,* [9]. We also note that accuracy and TP rates are higher than for new protocols, but that so are FP rates. In other words, few successful protocols lack both features, but having them is by itself no guarantee of success, *i.e.,* the features are correlated with success but far from causal.

5.2 Network Services Protocols

Network services protocols (S) have many aspects in common with A & T protocols, and therefore so do their results even if differences exist.

Feature	OR	p-value
Value for other prot.	22.00	0.009
Accuracy	83.33%	
Confusion Matrix	TP= 0.92 FN= 0.08 TN=0.67 FP= 0.33	

Table 4: New S protocols (18 RFCs)

Table 4 points again to the need for new S protocols to add value to other protocols if they are to succeed, and its odds ratio and significance are even stronger than for A & T protocols. This may be because network services' primary purpose is to facilitate network usage, so that offering easier access or added functionality to other protocols is of even greater importance.

Table 5 reports results for existing S protocols and includes again backward compatibility as a key feature; one now present in all successful protocols (TP = 1), though not by itself a guarantee of success (FP = 0.67). Security is, however, now absent; maybe because the smaller ecosystem of network services makes the adoption of security extensions "slightly" less challenging?

5.3 Network Control Plane Protocols

Network control plane (C) protocols differ from network services protocols primarily in that they target the network as opposed to network users. So while both

Feature	OR	p-value
Backwards Compatibility	16.41	0.011
Accuracy	70.37%	
Confusion Matrix	TP= 1 FN= 0 TN=0.33 FP= 0.67	

Table 5: Existing S protocols (27 RFCs)

share close ties to the network, their success can be affected by different features, as reported in Table 6.

Features	OR	p-value
Domain-wide value	18.33	0.037
Value for other prot.	6.60	0.076
Accuracy	78.57%	
Confusion Matrix	TP= 0.71 FN= 0.29 TN=0.86 FP= 0.14	

Table 6: C protocols (14 RFCs)

The table identifies "domain-wide value," *i.e.,* the protocol's ability to realize its full value once adopted in a given domain, as an important factor in a protocol's success. This aligns with our intuition that deploying protocols that "touch" network devices is easier when their scope is limited (to a domain). The second feature, "adding value to other protocols," is also consistent with the notion that network control functions that benefit other protocols should have an easier time being adopted. However, its high p-value together with the small number of protocols in this category, make it difficult to draw strong conclusions from its selection.

5.4 Network Routing Protocols

Network routing protocols (R) include intra- and inter-domain protocols. Given the small number of RFCs involved, a single set of results is again presented in Table 7 for all protocols in this category. Backward compatibility emerges again as a key feature, in part because there are few "new" protocols in this category. Another feature is "replacing another protocol," which is likely a reflection of the fact that most routing protocols have had multiple versions, with each new version replacing the previous one. Finally, "domain-wide value" is also identified as important but with a negative impact. Its selection is somewhat ambiguous and appears driven in part by the fact that a number of intra-domain extensions did not succeed. This may, however, change as we extend the number of RFCs under consideration. It may also be caused by "transient noise" in our labeling process[12]. For example, most protocols associated with IPv6 have been marked as "not successful" to reflect the fact that IPv6 itself has not (yet) succeeded. However, this situation may be changing [4, 5].

[12]Something that is unavoidable given that our sampling is punctual, and a protocol's success evolves over time.

Features	OR	p-value
Backwards compatibility	101.59	0.000
Replacing another protocol	80.02	0.003
Domain-wide value	0.077	0.010
Accuracy	72.41%	
Confusion Matrix	TP= 0.95 FN= 0.05 TN=0.30 FP= 0.70	

Table 7: R protocols (29 RFCs)

5.5 Network Data Plane Protocols

Table 8 reports results for network data plane protocols (D) and singles out "performance improvement," as correlated with success. This aligns with the intuition that performance is of utmost importance in the data plane, so that protocols that offer performance improvements stand a stronger chance of success.

Feature	OR	p-value
Performance Motivated	12.00	0.027
Accuracy	76.19%	
Confusion Matrix	TP= 0.50 FN= 0.50 TN=0.92 FP= 0.08	

Table 8: D protocols (21 RFCs)

6. LIMITATIONS & EXTENSIONS

The paper's results are preliminary and should be interpreted as such, even if most features identified as significant to a protocol's success exhibit a high level of statistical significance. First and foremost, the number of samples (RFCs) included in the analysis needs to be expanded. This is not so much to improve the classifier's accuracy, as there will always be external factors we won't be accounting for, but to lessen sensitivity to errors in feature characterization. Expanding our data set should also extend to protocols beyond standards track RFCs, *e.g.,* Informational or Experimental RFCs and expired Internet Drafts. This will allow us to include more protocols, many of which did not succeed. Adding RFCs also provides the opportunity for stronger validation by allowing distinct training and testing sets.

We believe this effort can offer useful insight into protocol design by leveraging the collective expertise embedded in the IETF review process.

7. REFERENCES

[1] Available at https://en.wikipedia.org/wiki/Logistic_regression.

[2] See http://www.jmp.com.

[3] See http://www.cs.waikato.ac.nz/ml/weka/.

[4] See https://www.stateoftheinternet.com/trends-visualizations-ipv6-adoption-ipv4-exhaustion-global-heat-map-network-country-growth-data.html.

[5] See https://www.google.com/intl/en/ipv6/statistics.html.

[6] R. Böhme. Internet protocol adoption: Learning from Bitcoin. In *Proc. IAB workshop on Internet Technology Adoption and Transition (ITAT)*, Cambridge, UK, December 2013. Position Paper.

[7] K. P. Burnham and D. R. Anderson. *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Science & Business Media, 2002.

[8] M. Kuhn and K. Johnson. *Applied predictive modeling*. Springer, 2013.

[9] A. Ozment and S. Schechter. Bootstrapping the adoption of Internet security protocols. In *Proc. WEIS*, Cambridge, UK, June 2006.

[10] D. Thaler and B. Aboba. What makes for a successful protocol? RFC 5218 (Informational), July 2008.

[11] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.