# Detecting Targeted Malicious Email

**Rohan M. Amin, Julie J.C.H. Ryan, and J. René van Dorp** | George Washington University

**Targeted malicious emails facilitate the exploitation of computer networks and the exfiltration of sensitive information. A new email-filtering technique focused on persistent threat and recipient-oriented features outperforms other available techniques.**

Unsolicited email is not only a nuisance but can be potentially dangerous. Methods to filter it out work fairly well with conventional unsolicited commercial email (also known as *spam*) or email soliciting personal information (also known as *phishing*), but they don't work as well with targeted malicious email (TME) that facilitates computer network exploitation. Current detection algorithms work well for spam and phishing because it's easy to detect mass-generated email sent to millions of addresses (for more information, see the "Related Work in Email Filtering" sidebar); it's possible to gather emails with similar characteristics and message content to probabilistically identify them. TME, on the other hand, targets single users or small groups in low volumes. It's tailored specifically to the target recipient and engineered to appear legitimate and trustworthy. If we rely on current conventional detection methods, TME goes undetected.

## The Targeted Malicious Email Challenge

A network defender encounters different classes of threat actors with varying intents and capabilities. Conventional computer network attacks exploit network-based listening services such as Web servers, whereas targeted attacks often leverage social engineering through vehicles such as email. Email is especially dangerous because nearly all organizations allow email to enter their networks.

In mid-2005, the UK National Infrastructure Security Co-ordination Centre[1] and the US Computer Emergency Response Team[2] issued technical alert bulletins about targeted, socially engineered emails that drop Trojans to exfiltrate sensitive information. The intrusions occurred over a significant period of time, evaded conventional firewall and antivirus capabilities, and enabled adversaries to harvest sensitive information. In 2007, various government agencies (including the US Departments of Defense, State, and Commerce) experienced intrusion attempts.[3] The US-China Economic and Security Review Commission's 2008 and 2009 reports to Congress summarize open source reporting of targeted attacks against US military, government, and contractor systems to collect sensitive information.[4] A report prepared for the US-China Economic and Security Review Commission profiled an advanced cyberintrusion and documented TME.[5]

In all of these examples, the threat actors weren't necessarily looking for immediate financial gain. For such advanced persistent threats, acquiring valuable information is the real intention. Although many victims of illegitimate email have money, only certain organizations have the type of valuable information that yields long-term strategic advantage. This level of targeting and sophistication suggests a patient threat actor with the resources to reconnoiter a target environment and craft emails relevant to the recipients, using email addresses, subject lines, and content tailored to entice recipients to open the message. The threat actors can then attach malicious files or Web links or repurpose previously sent email appended with malicious content. Clearly, threat actors can't use

 Copublished by the IEEE Computer and Reliability Societies 

this sort of advanced targeting on an Internet-wide scale: they're after specific information.

## Dataset Construction

Given TME's specific features and the failure of traditional filtering techniques to reliably detect it, we developed an alternative filtering procedure. Figure 1 outlines our process. We look at features of the email that other filtering techniques don't typically extract, classifying them as persistent threat and recipient-oriented features. We selected these features on the basis of our analysis of a large dataset of actual TME from a Fortune 500 company with more than 100,000 email users. We fully reviewed our intentions with the company's legal counsel and information security personnel. Although we conducted the research using actual data, we've sanitized all results to anonymize both the company and any users of the company's email system.

To the best of our knowledge, reliable, scalable, and automated TME detection isn't yet possible with commercially available tools. For other researchers looking to create similar datasets, we recommend partnering with organizations that are already manually identifying TME.

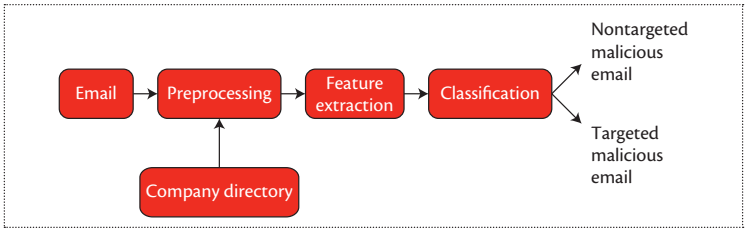


**Figure 1.** Classification process. A simplified view of our classification process first involves preprocessing email, leveraging company-specific information. Persistent threat and recipient-oriented features are extracted and the associated emails are classified using a random forest classifier.

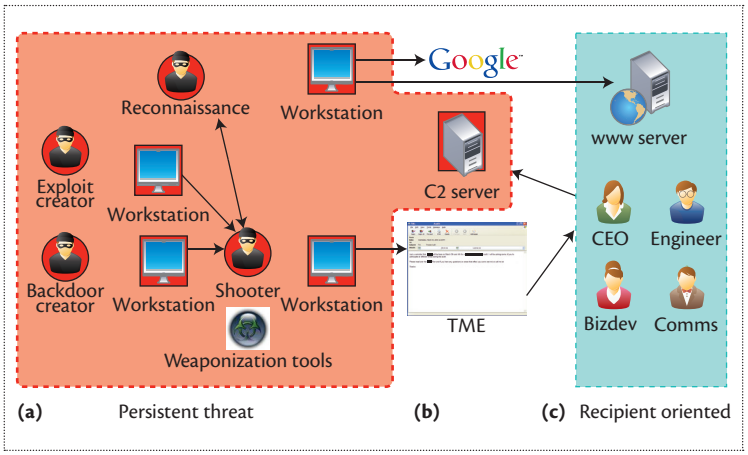


**Figure 2.** New features used for detecting targeted malicious email (TME). (a) The supply chain of components and threat actors necessary for creating a TME. (b) The Internet, which stands between the threat actors and email recipients. (c) TME recipients.

## Data

Typically, the datasets used to evaluate email-filtering techniques are incomplete or are an amalgamation of several different datasets. For example, the PU1 and ling-spam corpora, commonly used for evaluating the performance of spam filters, are made up of known spam and known legitimate emails from different sources.[6] Privacy concerns make it difficult to obtain legitimate email for analysis, and to further complicate matters, datasets sometimes lack email header information or are sanitized to the point where useful information is lost. Our study had to use full and complete emails, because a critical goal was to measure the added value of leveraging features of malicious email that are persistent threat and recipient oriented.

We leveraged complete emails from the company and additional recipient context, such as full name, job title, and business division membership. The complete dataset consists of three classes of emails: nontargeted malicious email (referred to as NTME1), targeted malicious email (referred to as TME1), and an evaluation set containing both TMEs and NTMEs (referred to as TS1). We used NTME1 and TME1 to construct the TME-filter technique and TS1 for evaluation. Figure 2 provides context for the new features we incorporated for TME detection.

**Nontargeted malicious email.** NTME1 consists of 20,894 randomly selected emails from 1 September through 20 November 2009. This dataset only includes emails from the Internet to the company (rather than emails that were only internal to the company) and only those emails that made it through a commercial antispam system. Thus, emails that the commercial antispam system classified as generic Internet spam weren't a part of this dataset. (Email in this dataset wasn't processed by a commercial antivirus system.)

**Targeted malicious email.** TME1 consists of a set of manually identified TMEs. Through manual computer forensics and information sharing with a community familiar with TME, the target company retroactively identified TME. We then manually reviewed these emails to confirm their proper classification. This dataset consists of 2,315 emails from 16 April through 19 December 2009, a longer time frame than the NTME1 period because TME isn't sent as often as NTME.

**Test set.** We created and used TS1 only for evaluation

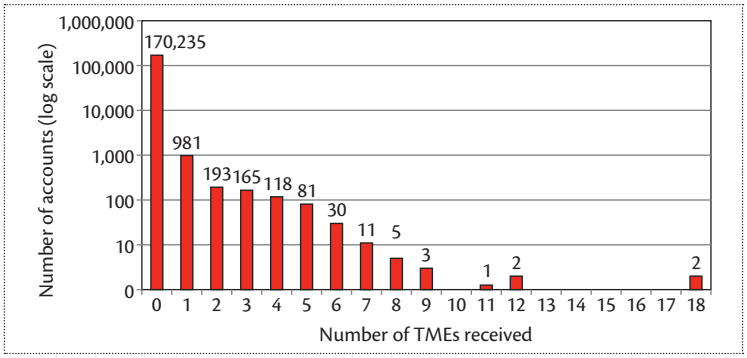


**Figure 3.** TMEs received by email accounts at the target company. Less than 1 percent of the email accounts at the company received any TME at all.

purposes. We didn't use it, as a whole or in parts, for analysis or training. The dataset consists of 1,457,729 emails from 22, 24, and 30 December 2009. We chose these three days because the company had retroactively discovered TME on those dates as a result of internal intelligence analysis and sharing with industry partners.

### Trends with TME and NTME

Separate analysis of TME informed the feature selection for our extraction purposes. Figure 3 shows the number of TMEs received by accounts at the target company: the vast majority of accounts received no TMEs, although a select set of accounts received more than one. Figure 4a shows the cumulative distribution function of Google search hits for the company's employees' email addresses, and Figure 4b shows the average amount of TMEs received by those email addresses sharing the same hit count in Google. The *x*-axis uses just the first 24 Google search hit bins, which accounts for 99.93 percent of the total population. Later bins are sparsely populated. We can see a positive correlation between these two variables.

Some TMEs appear to target employees with specific job titles. Table 1 shows the top 15 job titles in the target company by number of employees, NTMEs received, and TMEs received. It's interesting that the most common job title, systems engineering, isn't one of the top 15 TME recipients. Furthermore, the international business development job title, which consists of only 44 people in a company of more than 100,000 employees, is the third most targeted group.

## Persistent Threat and Recipient-Oriented Features

Figure 2 provides the context for the persistent threat and recipient-oriented features that we extracted to support the classification objective. We extracted a total of 83 features (detailed elsewhere[7]) and used them as input to a random forest classifier to construct the TME-filtering technique. Table 2 lists the top 10 of these 83 features.

### Persistent Threat

When threat actors *weaponize* (or embed malicious content in) an email, they can leave fingerprints useful for detection. Inevitably, threat actors, being human, resort to automation or other procedural techniques that can enhance detectability across several repeated intrusion attempts. To save on cost, threat actors might reuse weapons with different delivery vehicles. The combination of tools, techniques, and procedures measures capability. Both the tools used and locale are persistent threat features that we can incorporate in TME-filtering techniques.

Some automation tools actually leave names in an email, whereas other tools leave more subtle clues. For example, feature 10 in Table 2 is an artifact of certain email tools. When a threat actor is preparing and launching an email weapon, certain elements of his or her locale might be left in attachments or in the email itself. We can infer locale through language settings, character encodings, time-zone settings, Internet Protocol addresses, and system host names.

### Recipient Oriented

Threat actors might send emails to a particular individual because of his or her role in an organization. They might target the company's CEO, thinking his or her system might have sensitive information. Employees in business development might be prone to TME simply because their email addresses are more readily available as a function of their job; a senior-level employee might be more likely to be targeted than an entry-level employee.

Just as some sender-focused reputation techniques maintain lists of known bad senders, recipient reputation involves maintaining a list of recipients known to receive TME. It's conceivable that threat actors maintain a database of email addresses for a specific target organization and that these email addresses might receive a higher volume of TME over time. Feature 1 in Table 2 was only feasible as a recipient-oriented feature because the company keeps a log of those who have previously been targeted. Another dimension of reputation includes email visibility. Presumably, those email addresses that are more publicly available are more likely to be targeted. Email address visibility can be as straightforward as the number of times an email address appears in Internet search engine results. Furthermore, employees who have left a company might continue to receive TME to their no-longer-valid

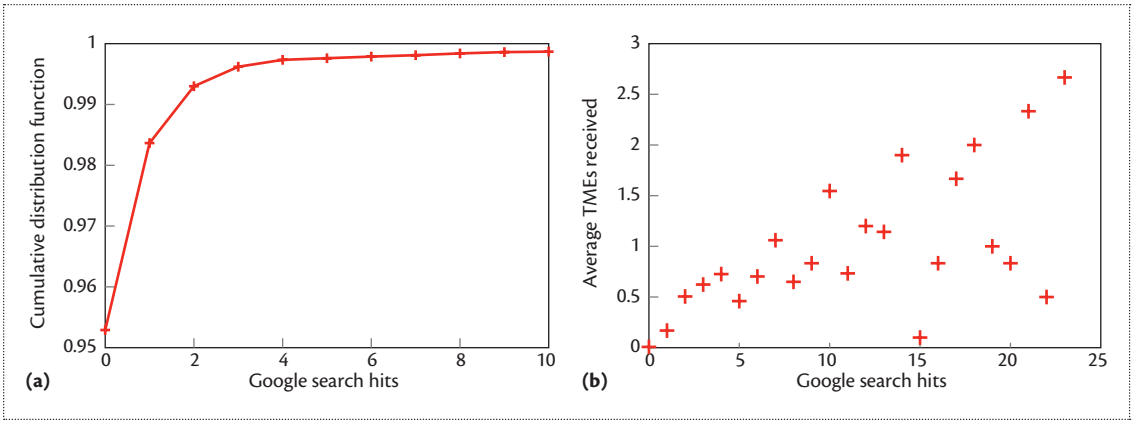


**Figure 4.** Analysis of Google search hits. (a) Cumulative distribution function for Google search hits. (b) Correlation between Google search hits and average TMEs received.

**Table 1. Top 15 job titles for number of employees, nontargeted malicious emails received, and targeted malicious emails received.**

| Rank | Number of employees | NTME received | TME received |
|---|---|---|---|
| 1 | Systems engineering | Program management | Business development analysis |
| 2 | Software engineering | Administrative assistant | Program management |
| 3 | Program management | Systems engineering | International business development |
| 4 | Embedded software engineering | Business development analysis | Communications |
| 5 | Mechanical engineering | Subcontract administrator | Business development |
| 6 | Member engineering staff | Procurement representative | Project specialist |
| 7 | Multifunctional finance | Project engineering | Mechanical engineering |
| 8 | Systems integration and test | Systems integration | Software engineering |
| 9 | Quality assurance | Business development | Fellow |
| 10 | Project engineering | Employment representative | Electronics engineering |
| 11 | Administrative assistant | IT program manager | Project engineering |
| 12 | Systems integration | Computer systems architect | Research engineering |
| 13 | Aeronautical engineering | Multifunctional finance | Communications representative |
| 14 | Systems administrator | Contracts negotiator | Research scientist |
| 15 | Electrical engineering | Member engineering staff | Field engineering |

email address because their email addresses still appear on websites or in threat actors' databases.

## Experimental Setup

We used the random forest classifier[8] to separate NTME from TME. Several characteristics of this classifier made it ideal for the datasets in this study:

- it can handle a large number of features;
- it can handle a large number of emails;
- it can handle a mixture of binary, numeric, and categorical features;
- it generally doesn't overfit;
- it can handle missing features;
- it trivially parallelizes the algorithm to scale up for huge datasets;
- it can estimate which features are more important than others; and
- it can handle unbalanced datasets (for example, a much greater number of NTMEs than TMEs).

**Table 2. Top 10 features as determined by random forest.**

| Rank | Feature | Type |
|---|---|---|
| 1 | Average number of TMEs received by envelope recipients | Recipient oriented (numeric) |
| 2 | Date header time zone | Persistent threat (categorical) |
| 3 | Outlook Express X-Mailer | Persistent threat (binary) |
| 4 | Email size | Persistent threat (numeric) |
| 5 | From domain = Gmail | Persistent threat (binary) |
| 6 | Average Google search count for envelope recipients | Recipient oriented (numeric) |
| 7 | Base64 character encoding | Persistent threat (binary) |
| 8 | GB2312 character encoding | Persistent threat (binary) |
| 9 | Link in email body to a ZIP file | Persistent threat (binary) |
| 10 | MIME boundary beginning with "2rfk" | Persistent threat (binary) |

**Table 3. Possible outcomes from the classifier.**

| | Actually TME | Actually NTME |
|---|---|---|
| Predicted as TME | True positive (TP) | False positive (FP) |
| Predicted as NTME | False negative (FN) | True negative (TN) |

Traditional decision-tree classification algorithms split each node using the *best split* from all available features. The best split is that which provides the most separation in the data. With random forests, each node splits (using the best split) from a randomly selected set of features at that node. In addition, they create multiple decision trees using bootstrap samples (random selections with replacements) from the dataset. These trees are created independently of each other and are classified according to a simple majority vote from the trees in the forest. The algorithm[8,9] is as follows:

1. In this study, trees grow to maximum size: $k$ = number of trees to create; $m$ = number of random features to select for node splitting; and $d$ = maximum depth of the trees.
2. Select $k$ vectors from the training data such that vector $\theta_k$ is chosen independent of $\theta_1, \ldots, \theta_{k-1}$.
3. For each of the bootstrap samples, grow a tree $T_k$, where each node splits using the best split from $m$ randomly selected features. The result is multiple tree classifiers $T_k : h(\mathbf{x}, \theta_k)$, where $\mathbf{x}$ is an input vector of unknown classification.
4. To classify $\mathbf{x}$, process that feature vector down each tree in the forest. Each tree will output a classification, also known as a *vote*. If $C_k(\mathbf{x})$ represents the classification of the $k$th tree in the forest, then the aggregate classification of the forest, $C_{forest}(\mathbf{x})$ = majority vote $\{C_k(\mathbf{x})\}_1^k$.

The 83 features extracted from email are represented as a vector of features. The output of the random forest classifier for a particular email is binary, classified as either TME or NTME using the email's specific vector of persistent threat and recipient-oriented features as input.

When the classifier correctly predicts a TME, it's a *true positive* (TP). When the classifier correctly predicts an NTME, it's a *true negative* (TN). When the classifier predicts an NTME as TME, it's a *false positive* (FP) or Type I error. When the classifier predicts a TME as NTME, it's a *false negative* (FN) or Type II error. Table 3 shows the possible outcomes from the classifier.

The false positive rate (FPR) is the proportion of NTME that was incorrectly classified as TME. The specificity is equal to 1 – FPR, where the FPR is

$$\frac{\text{FP}}{\text{FP+TN}}.$$

The false negative rate (FNR) is the proportion of TME that was incorrectly classified as NTME. The sensitivity is equal to 1 – FNR, where the FNR is

$$\frac{\text{FN}}{\text{FN+TP}}.$$

False positives and false negatives aren't of equal consequence. In the case of spam or phishing, users generally don't want to miss legitimate email and therefore are willing to endure the annoyance of a few misclassified spam messages in their inbox. In this scenario, the

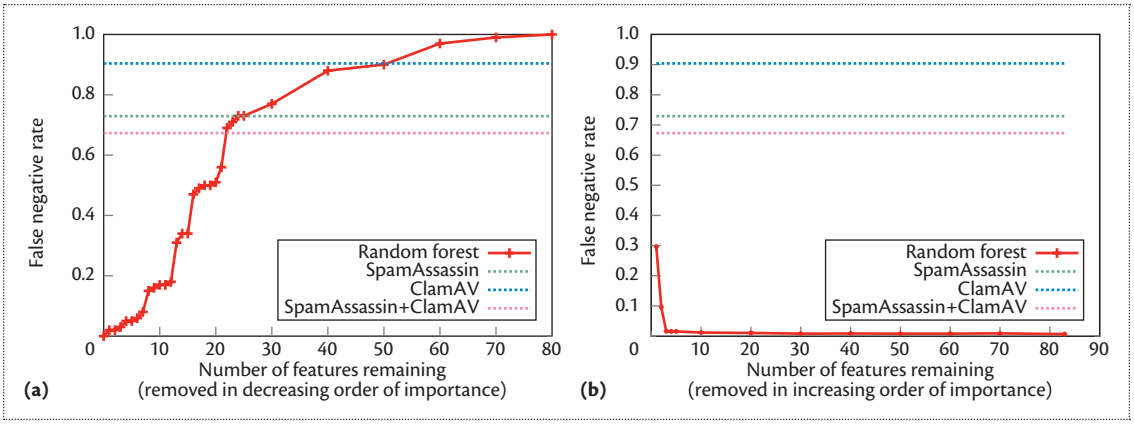


**Figure 5.** Feature importance analysis for the NTME1–TME1 dataset. (a) Removing the most important features first. (b) Removing the least important features first.

**Table 4. Contingency table for TME detection, NTME1–TME1.**

| SpamAssassin+ClamAV results | Random forest results | | |
|---|---|---|---|
| | Correct | Error | Total |
| Correct | 742 | 5 | 747 |
| Error | 1,558 | 10 | 1,568 |
| Total | 2,300 | 15 | 2,315 |

cost of a false positive is greater than a false negative. In the case of TME, the costs are reversed. A false negative means TME was misclassified as NTME. Given that TME is associated with advanced threats, missing a single TME from a detection standpoint can result in threat actor presence on a network and significant impact to the targeted organization.

## Results

Recall the construction of the NTME1, TME1, and TS1 datasets. At first, we used a 10-fold cross validation as our evaluation method for the joint NTME1–TME1 dataset. Later, we used the joint NTME1–TME1 dataset for training, but instead of doing cross validation, we used the independent TS1 dataset to evaluate the TME filter constructed using the joint NTME1–TME1 dataset.

### NTME1–TME1 Dataset

We started the analysis of the NTME1–TME1 dataset by using the random forest classifier's measure of feature importance. A random forest using all 83 features with $k = 50$ and $m = 30$ produced the best result—an FNR of 0.6 percent. (Recall that false negatives are of greater consequence than false positives.)

Processing the NTME1–TME1 dataset using SpamAssassin (http://spamassassin.apache.org) results in an FNR of 73 percent, indicating that a large volume of TME evades SpamAssassin. ClamAV (www.clamav.net) has even poorer performance with a 90 percent FNR. Serializing both SpamAssassin and ClamAV (SpamAssassin+ClamAV) resulted in a 67 percent FNR.

Figure 5a shows the top 20 features (in order of feature importance) that have to be removed from the random forest classifier before its FNR approaches that of SpamAssassin+ClamAV. Figure 5b presents analysis results following a similar process, but it removes the least important features first. Even with only one feature remaining (removed in order of increasing feature importance), the random forest classifier outperforms SpamAssassin and ClamAV.

A McNemar test, summarized in Table 4, comparing the random forest classifier against SpamAssassin+ClamAV, yields a $\chi^2$ test statistic of 1,541.1, which is greater than the critical value of 6.635 at the $\alpha = 0.01$ level of significance.

We must reject the null hypothesis that the two detection methods are the same in their abilities to detect TME. Table 4 shows that the random forest–based TME filter technique developed herein identifies 2,300 out of 2,315 TMEs correctly (99 percent),

**Table 5. Evaluation contingency table for TME detection, TS1.**

| SpamAssassin+ClamAV results | Random forest results | | |
|---|---|---|---|
| | Correct | Error | Total |
| Correct | 7 | 0 | 7 |
| Error | 33 | 4 | 37 |
| Total | 40 | 4 | 44 |

whereas the traditional SpamAssassin+ClamAV approach only identifies 747 out of 2,315 TMEs correctly (32 percent). Significant false positives were not introduced either: the random forest classifier had an FPR of 0.1 percent.

### TS1 Dataset

For the TS1 dataset, we trained the random forest classifier using the NTME1–TME1 dataset and then tested it using the TS1 dataset. A random forest with $k = 100$ and $m = 2$ produced the best results.

Processing the TS1 dataset using SpamAssassin results in an FNR of 89 percent; ClamAV has a slightly better performance with an 84 percent FNR. Serializing both SpamAssassin and ClamAV yields no additional benefit with an 84 percent FNR. Processing the TS1 dataset using a random forest classifier trained with the NTME1–TME1 dataset yields an FPR of 9.1 percent.

A McNemar test, summarized in Table 5, comparing the random forest classifier against SpamAssassin+ClamAV, yields a $\chi^2$ test statistic of 31.03, which is greater than the critical value of 6.635 at the $\alpha = 0.01$ level of significance. Therefore, we reject the null hypothesis that the two detection methods have the same ability to detect TME. Table 5 shows that the random forest-based TME filter technique we developed classifies 40 out of 44 TMEs correctly (91 percent), whereas the traditional SpamAssassin+ClamAV approach only identifies 7 out of 44 TMEs correctly (16 percent). In addition, the FPR for the random forest, 0.009 percent, was negligible. Although detection performance is worse for both methods with the TS1 dataset, the random forest method with persistent threat and recipient-oriented features clearly outperforms SpamAssassin, ClamAV, and the joint SpamAssassin+ClamAV filter.

For future research, we hope to extend feature extraction to file attachment metadata. Threat actors might inadvertently leave remnants of information such as file paths, time zones, or even author names.[10] All these features might associate multiple intrusion attempts into a related campaign. In addition, organizations can track features that characterize the types and amounts of email received by a particular email address. For example, for each recipient, the number of emails and attachments received over a fixed time period might help uncover email that falls outside of his or her normal receiving patterns. For email with hyperlinks, we could develop features to indicate whether the domain of a link has ever been visited before. We could also incorporate information related to domain creation. Aside from extending email classification features, we could also map features to different threat actors for a multiclassification model. As organization and recipient-oriented information evolves, we hope to evolve our techniques accordingly. ■

### References


1. *Targeted Trojan Email Attacks*, briefing 08/2005, Nat'l Infrastructure Security Co-ordination Centre, 2005; www.egovmonitor.com/reports/rep11599.pdf.
2. *Targeted Trojan Email Attacks*, tech. cybersecurity alert TA05-189A, US-CERT, 2005; www.us-cert.gov/cas/techalerts/TA05-189A.html.
3. J.A. Lewis, "Holistic Approaches to Cybersecurity to Enable Network Centric Operations," statement before Armed Services Committee, Subcommittee on Terrorism, Unconventional Threats and Capabilities, 110th Cong., 2nd sess., 1 April 2008.
4. *2009 Report to Congress of the U.S.-China Economic and Security Review Commission*, report, Nov. 2009; www.uscc.gov/annual_report/2009/annual_report_full_09.pdf.
5. B. Krekel, *Capability of the People's Republic of China to Conduct Cyber Warfare and Computer Network Exploitation*, Oct. 2009; www.uscc.gov/researchpapers/2009/NorthropGrumman_PRC_Cyber_Paper_FINAL_Approved%20Report_16Oct2009.pdf.
6. I. Androutsopoulos et al., "An Experimental Comparison of Naive Bayesian and Keyword-Based Anti-Spam Filtering with Personal E-mail Messages," *Proc. 23rd Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval*, ACM, 2000, pp. 160–167.
7. R.M. Amin, "Detecting Targeted Malicious Email through Supervised Classification of Persistent Threat

## Related Work in Email Filtering

Much of the current research in email filtering focuses on high-volume, Internet-wide email abuse. These filtering techniques fit into five classes:

- *Authentication techniques* typically relate to the Domain Name System.[1]
- *Contextual techniques*, often used for spam-filtering, typically leverage Bayesian approaches.[2]
- *Characterization techniques* typically focus on network traffic behavior. Robert Beverly and Karen Sollins exploited the fact that spammers need to leverage large numbers of resource-constrained hosts.[3] Network transport layer properties (such as packet round-trip time) uncover these hosts, which are often found on residential Internet connections.
- *Reputation-filtering techniques* leverage lists of good and bad elements to calculate a level of trust. For example, David Erickson and his colleagues used a combination of challenge responses to validate unknown senders and a persistent white list per user for filtering legitimate email.[4]
- *Resource-consumption techniques* actively increase costs to senders through increased use of network bandwidth or computing power. For example, Minh Tran and Grenville Armitage introduced additional latency to network flows associated with spam.[5] By doing this, connections take longer to complete and result in senders needing more resources to send the same quantity of email (as long as no delays are introduced).

Nearly all current email-filtering techniques are effective only when applied to large volumes of unwanted email, but they don't work for TME, where the emails of interest are low in volume. A more complete survey of current email-filtering techniques can be found in "Detecting Targeted Malicious Email through Supervised Classification of Persistent Threat and Recipient Oriented Features."[6]

**References**


1. M. Wong and W. Schlitt, "Sender Policy Framework (SPF) for Authorizing User of Domains in E-Mail," tech. memo, Internet Soc., 2006; www.ietf.org/rfc/rfc4408.txt.
2. M. Sahami et al., *A Bayesian Approach to Filtering Junk Email*, tech. report WS-98-05, Am. Assoc. Artificial Intelligence, 1998.
3. R. Beverly and K. Sollins, *Exploiting Transport-Level Characteristics of Spam*, tech. report MIT-CSAIL-TR-2008-008, Computer Science and Artificial Intelligence Lab, MIT, 2008.
4. D. Erickson, M. Casado, and N. McKeown, "The Effectiveness of Whitelisting: A User-Study," *Proc. Conf. Email and Anti-Spam*, 2008; www.ceas.cc/2008/papers/ceas2008-paper-20.pdf.
5. M. Tran and G. Armitage, "Evaluating the Use of Spam-Triggered TCP Rate Control to Protect SMTP Servers," *Proc. Australian Telecom. Networks and Applications Conf.* (ATNAC 04), ATNAC, 2004, pp. 329–335.
6. R.M. Amin, "Detecting Targeted Malicious Email through Supervised Classification of Persistent Threat and Recipient Oriented Features," PhD thesis, George Washington Univ., 2011.



and Recipient Oriented Features," PhD thesis, Dept. Eng. and Applied Sciences, George Washington Univ., 2011.
8. L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, 2001, pp. 5–32.
9. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer, 2008.
10. E. Hutchins, M. Cloppert, and R. Amin, "Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains," *Proc. 6th Int'l Conf. Information Warfare and Security* (ICIW 11), Academic Conferences, 2011, pp. 113–125.



**Rohan M. Amin** is a program director with a large defense contractor. His research interests include advanced email filtering, threat actor analysis, and network intrusion reconstruction. Amin has a PhD in engineering management from George Washington University. Contact him at rma@gwu.edu.

**Julie J.C.H. Ryan** is an associate professor and chair of the Department of Engineering Management and Systems Engineering at George Washington University. Her research interests include information security and information warfare research. Ryan has a DSc in engineering management from George Washington University. Contact her at jjchryan@gwu.edu.

**J. René van Dorp** is a professor in the Department of Engineering Management and Systems Engineering at George Washington University. His research interests include uncertainty analysis, distribution theory, risk management analysis, probabilistic risk assessment, and reliability. Van Dorp has a DSc in operations research from George Washington University. Contact him at dorpjr@gwu.edu.