

## Practical Threat Analysis for the Software Industry

by: Ygor Goldberg, 01/10/2005

<http://www.securitydocs.com/library/2848>

*"The Dark Tower had been rebuilt, it was said. From there the power was spreading far and wide, and away far east and south there were wars and growing fear. Orcs were multiplying again in the mountains. Trolls were abroad, no longer dull-witted, but cunning and armed with dreadful weapons. And there were murmured hints of creatures more terrible than all these, but they had no name."*

J.R.R. Tolkien. The Lord of the Rings

### Preface

This paper describes Practical Threat Analysis (PTA); a calculative threat modeling methodology and a CASE tool that assists software security analysts and software developers in assessing system risks and building the most effective risk reduction policy for their system.

### What is threat analysis?

Threat analysis identifies threats and defines a cost-effective risk mitigation policy for a specific architecture, functionality and configuration. It involves the mapping of assets, modeling of threats and building of a mitigation plan that lowers system risk to an acceptable level. The mitigation plan is composed of countermeasures which are considered to be effective against the identified threats.

### When should threat analysis be applied?

Threat analysis is required for:

- Complex software systems that integrate multiple infrastructures and technologies.
- Customized application solutions built on standard products.
- All other cases where it is unacceptable to implement pre-compiled "to-do" lists provided by a software vendor or standards committee.

Threat analysis should be used in the earliest possible stages of system design and thereafter as an ongoing process throughout the system's lifecycle of development, integration, change requests and problem management.

### The problem

Software development is always constrained by some combination of budget, time and resources and threat analysis usually ends up as a task to be done "later". Threat analysis is a skill most programmers and managers lack which results in the task being done "never".

### The solution

By using PTA, analysts who are expert in the application domain can quickly build and analyze risk management models and policies without endangering the project schedule. Knowledge is retained, shared and maintained within the group and program management has total transparency to system risk without the need for additional resources.

### What are the existing tools?

Word-Processor + Spreadsheet Documents – The analyst has the freedom to describe threats and

vulnerabilities and express her analytical qualification in a free format with no restrictions dictated by the tool. However, the overhead of the data management and the calculation tasks is very high because of the lack of a built-in ability to represent the interrelations between entities and to dynamically alter the threat model. In reality the data model required for threat modeling is far beyond the capabilities of spreadsheet programs. In addition, most of these solutions also lack the necessary reporting functionality.

Checklist-Based Tools – These are tools that provide pre-defined sets of security recommendations that are used as checklists. This approach may work for standard applications where all possible security issues are known in advance. Most of these tools have reporting capabilities and usually come in two flavors:

- Questionnaire-based[1] in which the user is asked to answer a series of questions that reflect the embedded checklist.
- Template-based[2] in which the user is asked to distinguish the specifics of her application from the standard checklist.

Since this type of tool is based on lists of general purpose standard countermeasures, they are not flexible in supporting and encouraging the analyst to create new threat scenarios that are specific to her application.

Threat Modeling Tools – Microsoft's[3] tool combines Schneier's Attack-Trees methodology[4] with standard Microsoft Threat Classification[5] and has four important limitations:

- Doesn't relate threats to financial losses caused by the attacks and does not rank countermeasures by their effectiveness and priority in reducing risk.
- Uses "pre-defined" cases and doesn't easily fit application-specific threat scenarios
- Doesn't provide a complete system view for threat analysis risk management.
- Limited reporting and collaborative capabilities

## **Introducing PTA**

The PTA calculative methodology\* and CASE tool enable effective management of operational and security risks in complex software systems by an existing team. It provides an easy way to maintain dynamic threat models that are capable of reacting to changes in the system's assets and vulnerabilities. *With PTA an analyst can maintain a growing database of threats, create documentation for security reviews and produce reports showing the importance of various threats and the priorities of the corresponding countermeasures.*

PTA automatically recalculates threats and countermeasures priorities and provides decision makers with updated action item lists which reflect the changes in threat realities. *Countermeasure priorities are expressed as a function of the system's assets values, degrees of damage, threat probabilities and degrees of mitigation provided by countermeasures to the threats.*

A software development team uses PTA from day one of design and throughout the system's lifecycle. PTA provides intuitive and easy ways for iterative interaction between threat analysts and developers. It supports a collaborative process of evaluating threats risks and ranking the cost-effectiveness of proposed countermeasures. *The team's "threat analyst" can be the program/product manager, system architect or development lead who can start being productive with the CASE tool within hours.*

\* patent pending

## How does PTA relate to security standards?

How does PTA relate to security standards and initiatives, such as ISO17799, BS 7799–2002, SSE-CMM, Octave, FIPS 199, GAISP, COBIT and others?

PTA complements existing standards and appraisal procedures by supplying means for the actual definition of threats, vulnerabilities and proposed countermeasures. It manages a well designed database of all relevant security entities and enables production of documentation for the evaluation procedures required by the standards.

Standards recommend procedures for organizations to follow in order to ensure information systems security. These recommendations include mapping of assets, vulnerabilities, threats and countermeasures, assessment of risks and implementation of risk mitigation plans. PTA provides the actual means for performing these tasks in a productive way.

Some standards provide lists of numerous recommended countermeasures that should be implemented. These lists may serve the analyst as a baseline of definitions of common vulnerabilities and countermeasures and can help him in grasping the terminology. PTA enables the integration of these entities in its database. However it should be noted here that the standard lists cannot cover the most intimate aspects of customized solutions and the specifics of complex systems that integrate several technologies. At best, compliance with standards provides only the baseline security and additional analysis of application-specific risks is required.

PTA may also serve as the foundation of Information Security Management System - a concept that is promoted by modern standards. Its growing database and statistics may be used as an evidence of the organization's efforts for constantly improving the process of threat and vulnerability analysis.

## Terminology

### System

System is a cluster of software modules and hardware components together with sets of operational and business procedures that are the target of the threat analysis process. Systems are characterized by their specific goals, functionality, architecture, configuration and users.

System's Maximal Risk is a calculated value that expresses the maximal loss that may be caused to the system's assets due to the threats that were identified. It reflects the potential risks of all threats to the system's assets and is displayed both in \$ value as well as in percentage of the total system's assets.

System's Minimal Risk is a calculated value that expresses the loss that may be caused to the system's assets after all the countermeasures in mitigation plans are implemented. It reflects the remaining risks of all threats after full implementation of all the mitigation plans and is the actual lowest value of risk that can be achieved. It is displayed both in \$ value as well as in percentage of the total system's assets.

System's Current Risk is a calculated value that expresses the loss that may be caused to the system's assets according to current implementation level of mitigation plans. It is displayed both in \$ value as well as in percentage of the total system's assets.

System's Total Value of Assets is the calculated total value of all the system's assets.

System's Countermeasures Implementation Cost is the calculated value of the cost of the implementation of the countermeasures in all mitigation plans.

System's Current Investment in Implementation is the cost of the implementation of the countermeasures that are already applied to the system.

## **Vulnerability**

Vulnerability is a weakness, limitation or a defect in one or more of the system's elements that can be exploited to disrupt the normal functionality of the system. The weakness or defect may be either in specific modules of the system, its layout, its users, operators, and/or in its associated regulations, operational and business procedures.

## **Countermeasure**

Countermeasure is a procedure, action or mean for mitigating a specific vulnerability. A specific countermeasure may mitigate several different vulnerabilities. In some standards documentation, countermeasures are called "controls" or "safeguards".

Countermeasure's Fixed Cost is the estimated value (in \$) of the one-time expense associated with the implementation of the countermeasure, e.g. purchase of equipment, enhancing the software, etc.

Countermeasure's Fixed Cost Period is the number of years over which the fixed cost expense lasts (both from economical aspects as well as from book-accounting considerations).

Countermeasure's Recurring Cost is the estimated recurring expense (in \$) that derives from applying the countermeasure, e.g. administrator's salary, insurance payments etc.

Countermeasure's Weighted Cost is the calculated weighted average of the countermeasure's fixed and recurring implementation costs and is displayed in 'annual \$' units.

Countermeasure's Overall Mitigation is the calculated degree of mitigation provided by a specific countermeasure to the overall risk of the system and is displayed as in percentage of the overall risk.

Countermeasure's Cost-Effectiveness is the degree of mitigation provided by a specific countermeasure to the overall risk in the system in relation with the cost of implementing this specific countermeasure. The value is displayed in "percents of overall mitigation per 1,000 \$" units. Note that the countermeasure's cost-effectiveness does not take into consideration the countermeasures which are already implemented therefore it is not necessarily identical to the practical PTA recommendation on the countermeasures that should be implemented in order to reduce the system's risk.

## **Asset**

Asset is information, capability, an advantage, a feature, a financial or a technical resource that may be damaged, lost or disrupted. Assets may be digital (software sources), physical (a server machine) or commercial (the corporate brand). The damage to an asset may affect the normal functionality of the system as well as of the individuals and/or organizations involved with the system.

Asset's Fixed Value is the estimated value (in \$) of the one-time expense associated with the loss of the asset, e.g. the value of the loss caused by blocking the company's e-commerce operation for 7 days etc.

Asset's Fixed Value Period is the number of years over which the fixed value expense lasts (both from economical aspects as well as from book-accounting considerations).

Asset's Recurring Value is the estimated recurring value (in \$) of losses that may be caused when the asset is damaged e.g. recurring expense due to the non-availability of a software service.

Asset's Weighted Value is the calculated financial value of the loss when asset is totally damaged, destroyed or stolen. The value is displayed in 'annual \$' and expresses the weighted average of the asset's fixed and recurring values in \$ per year units.

Asset's Relative Value is the calculated percentage of the specific asset's value from the total value of all the system's assets.

Asset's Maximal Risk is the calculated maximal risk (in percentage of the asset's value) that threatens the asset. The calculation is based on the parameters of all threats that might damage the asset.

Asset's Minimal Risk is the calculated risk that threatens the asset after all mitigation plans are implemented. It reflects the actual lowest value of risk that can be achieved after the full implementation of all mitigation plans of the threats that threaten the asset.

Asset's Current Risk is the calculated risk that threatens the asset according to current implementation level of mitigation plans.

## **Threat**

Threat is a specific scenario or a sequence of actions that exploits a set of vulnerabilities and may cause damage to one or more of the system's assets.

Threat's Probability is the likelihood that the threat scenario will materialize. In some documentation the threat's probability is characterized by the term "Annual Occurrence Rate" (AOR).

Threat's Damage Level to Asset is the financial value of damage caused by a specific threat to a specific asset expressed in percentage of the asset's value - if level is 100% the damage to the asset is maximal.

Threat's Damage is the total damage (in percentage of the total value of all assets) that the threat may cause to the system. The calculation is based on the damage caused to each of the assets threatened by the threat.

Threat's Maximal Risk is a calculated value that expresses the maximal loss that may be caused to the system's assets due to the specific threat. It reflects the potential risk of the threat to the system's assets and is displayed both in \$ value as well as in percentage of the total system's assets. In some documentation the threat's risk is called "Annual Loss Expectancy" (ALE).

Threat's Minimal Risk is a calculated value that expresses the loss that may be caused to the system's assets after all the countermeasures in mitigation plan of the specific threat are implemented. It reflects the actual lowest value of risk that can be achieved after the full implementation of all mitigation plans of the threat and is displayed both in \$ value as well as in percentage of the total system's assets.

Threat's Current Risk is a calculated value that expresses the loss that may be caused to the system's assets according to current implementation level of the threat's mitigation plan. It is displayed both in \$ value as well as in percentage of the total system's assets.

Threat's Recommended Countermeasures is a set of all possible countermeasures that may mitigate the threat and reduce the threat's risk. This set is based on the countermeasures that mitigate the threat's vulnerabilities.

Threat's Mitigation Plan is a subset of threat's recommended countermeasures that is assumed to be the most effective for mitigating a specific threat. The decision which of the recommended countermeasures will be included in the Threat's Mitigation Plan is made by the analyst, who uses his/her expertise to decide which countermeasures are most effective when applied together.

Threat's Countermeasure Mitigation Level is the mitigation level that a specific countermeasure would provide to a specific threat if it was the only countermeasure in the mitigation plan. It is displayed in percentage of the threat's overall risk.

Threat's Maximal Mitigation is the maximal mitigation level (in percentage of the specific threat's risk) that may be achieved by applying all countermeasures in threat's mitigation plan.

Threat's Current Mitigation is the portion of mitigation (in percentage of the specific threat's risk) that is provided by the countermeasures that are already implemented.

### The Threat Model

The following scheme describes the interrelations between a threat and the assets, vulnerabilities and countermeasures entities.

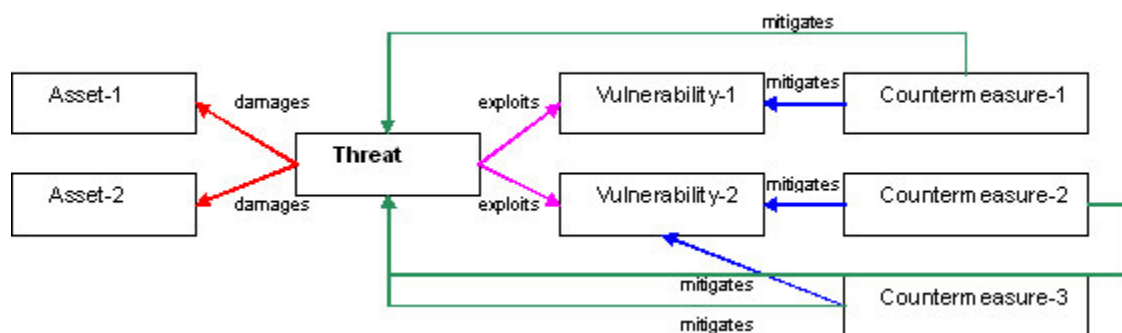


Figure 1: PTA data model sample scheme

The threat described in Figure 1, causes damage to Asset-1 and Asset-2 and exploits two vulnerabilities: Vulnerability-1 and Vulnerability-2. Vulnerability-1 is mitigated by Countermeasure-1 and Vulnerability-2 is mitigated by Countermeasure-2 and Countermeasure-3 as noted by the blue arrows. Since a threat may exploit several vulnerabilities, the set of possible countermeasures that might mitigate a threat is completely defined by the set of vulnerabilities used in a threat scenario and is noted by the green arrows in the scheme.

### Attacker Type

Attacker is a person (or group of people) that may perform the steps of a specific threat scenario and attack the system's assets.

Attacker Types are the various classes of attackers differentiated by their motivation, qualification, available attack tools and their accessibility to the attacked system's resources e.g. hackers, insiders, users etc.

### **Entry Point**

Entry Point is a "door", either in the system itself or in the human operation associated with it that is used by attackers to penetrate the system, e.g. Web site, IVR service, SMS server, CRM representatives called by customers over the phone etc. The attacker may use several entry points for materializing a specific threat.

### **Tag**

Tag is a free text descriptive attribute that might be associated with assets, threats, vulnerabilities and countermeasures. Tags are often used for helping the analyst in classifying the various entities in the threat model and improving their comprehensibility.

### **Attached Document**

Attached Document contains additional unstructured information relevant to the threat analysis entities and process e.g. security notes, standards specifications, development ideas, design schemes etc. Attached documents may be associated with specific assets, vulnerabilities, countermeasures and threats at any step in the threat analysis process.

## **Threat Analysis Steps**

### **Prerequisites**

The threat analyst identifies system vulnerabilities, predicts even the most hypothetical threat scenarios and evaluates threat probability and risk to enable prioritizing the corresponding countermeasures.

Before starting out, the analyst should learn the system's terminology, functionality and architecture. The in-depth understanding of the system is of crucial importance for the correct identification of system vulnerabilities and the building of possible threat scenarios.

The following documentation is needed:

- Terminology dictionary that explains the terms and acronyms relevant to the system being analyzed
- Functional description of the system including all typical use cases
- Architectural diagram of the system and documentation for various system modules

These documents must be detailed enough to be used as reference for the decisions regarding the applicability of various threat scenarios to the analyzed system.

### **Preparing a List of Tags**

It is a good idea to prepare a list of relevant tags that will help the analyst in classifying the various threat model entities according to their specific properties e.g. define tags for each of the system's components or tags which describe the various areas in the system's architecture.

The tags can later be associated with the entities and improve the readability of the threat model.

## **Identifying System Assets**

The correct mapping of assets, their financial value and the evaluation of financial loss to the system's owner when these assets are damaged or stolen, is one of the most critical tasks in the threat analysis process. The assets value is used as the basis for calculating threat risks and countermeasures priorities.

An analyst may at times hear claims like “everything we have is important”. While this could be true for some systems, we believe it is not the typical case. It is more likely that assets need to be clearly prioritized. Consider, for example the following partial list of the assets of a financial institution:

- Office equipment such as printers
- Confidential information about institution's clients
- Clients' money
- Private keys used for authentication of transactions
- Master key used for generating private keys

The accurate assessment of the financial value of the damage that may be caused by losing each of the above assets will enable the correct classification of assets according to their importance to the institution and help avoid a situation where the institution invests resources in protecting printers while leaving the master key unprotected.

In some cases the value of assets is less intuitive especially when they are intangible. For example, the confidence of the public in an electronic trading system may be damaged by the appearance of non-relevant texts on the system's Web site. No money is lost, no information is disclosed, all technical resources are still functioning but the site reputation and the trust of the shoppers are shaken. An indirect financial loss should be set for this type of damage.

Due to the importance of asset mapping, we recommend that the asset list and corresponding values be periodically checked by non IT personnel e.g. the company's CFO, marketing officers and legal consultants. Analysts can quickly do a “what-if” analysis by modifying asset values and obtaining an insight on the model's accuracy and completeness.

In practice, it is often easier for the analyst to identify system assets via the process of analyzing specific threats (as described in the following). A fact of human nature is that we don't realize how valuable things are until we lose them. This implies an iterative approach of mapping assets and threats.

### **Identifying System Vulnerabilities – the real ones**

Identifying vulnerabilities requires the analyst to be intimate with the system's functionality, architecture, implementation and deployment details. The analyst should also be familiar with business and operational procedures and the types of users and other parties involved in system operation.

An analyst can use the Web to find generally known vulnerabilities as published by software vendors and security consultants. Most of the items in these check lists are, in many cases, irrelevant to the specific system or may be easily solved by a simple comprehensive routine such as “always install most updated vendor's security patches”. The thing that should concern us here is that such a list will draw the attention of the analyst away from the real vulnerabilities that are specific to the system which is being analyzed.

Therefore we highly recommend that the analyst should investigate the system's architecture and implementation details and collaborate with architects, developers, installers and support engineers as well as with the business managers of the system to discover the real vulnerabilities that are unique to the system and that may not be identified without this intimate knowledge. *From experience – the most severe vulnerabilities reside in the interfaces, junctures and stitches between the various elements in*

*complex systems and rarely appear in the standard lists.*

As mentioned before, the identification of the relevant vulnerabilities is a continuous iterative task bundled with the step of identifying threats (described below) – the real sophisticated vulnerabilities are identified when building threat scenarios.

### **Identifying Countermeasures**

Identifying countermeasures has two outputs:

- A list of countermeasures that protect vulnerabilities. The list includes the implementation cost of each countermeasure and the countermeasure's relevant tags. If the countermeasure is already applied it should be marked as 'already implemented' to enable producing updated statistics of the current risk level in the system.
- A map of the relationships between countermeasures and vulnerabilities. This map shows which vulnerability may be mitigated by a specific countermeasure. Sometimes a countermeasure is introduced as a solution to a specific vulnerability, but after additional consideration it turns out that it may help in mitigating other vulnerabilities too.

The accurate identification of countermeasures and their relations with vulnerabilities is the basis for building risk mitigation plans as described in the next steps.

### **Classification of Potential Attacker Types**

Classification of the relevant attacker types may be helpful in focusing the analysis on practical realities. The classification of attackers is useful when we can clearly relate each of the threats with one or more of the attacker types.

Attacker type's data includes the understanding of his/her motivation as well as his qualification, available attack tools and her accessibility to the system's resources. Special care should be given to the classification of 'insiders' attacker type since their activity may be very dangerous.

A good starting point can be defining an attacker type for each of the user roles which appear in the system's use cases and reserve few more attacker types to hackers and other types of bandits.

### **Identifying Potential Entry Points**

The best tactic for this step is to review the list of attacker types and document every possible way the potential attackers could access the system. The list of entry points may be revisited and clarified while analyzing threats.

### **Building Threat Scenarios and Mitigation Plans**

This is the most important step of the threat analysis process. Its outcomes are:

- A list of the system's threats
- A map of the relationships between threats and tags, assets, attacker types, entry points and vulnerabilities
- An evaluation of the total damage and risk parameters for each of the threats
- Mitigation plans and the evaluation of the remaining system's risk data

Since threats are the most complex entities in the model, the process of identifying and constructing the threat's elements and parameters has a 'decomposition' nature. During this process the analyst will have to return to previous analysis steps in order to create missing entities, such as assets and vulnerabilities referred by the threat which is constructed. The following describes the sub-steps of building a threat scenario and a mitigation plan for a single threat.

#### Initializing threat - start from name and description

Always start by giving the threat a name and a short textual description (a few sentences) which includes the actions taken by the attacker and the description of the impact the threat has on the system. The description will be used as reference for the following steps and will be refined during the process.

#### Identifying the damaged assets and the damage levels

Identify the list of assets that may be damaged by the threat and the maximal damage level that may be caused by the threat to each of the assets. That is needed in order to enable the automatic calculation of the total damage (the financial losses) that may be caused to the system if the threat materializes.

#### Setting the threat's probability

Threat's probability is the likelihood of the occurrence of a specific threat scenario at least once a year. Its value may be in a range of 0 - 1 where 0 means that threat will never materialize and 1 means that threat will definitely materialize at least once a year. The threat's risk value is automatically calculated based on the threat's total damage and the threat's probability.

#### Identifying the exploited vulnerabilities

The correct identification of the vulnerabilities exploited by the threat scenario is important for building the threat's mitigation plan since the identified vulnerabilities define the collection of proposed countermeasures available for mitigating the threat. When vulnerabilities are identified, the list of proposed countermeasures is populated automatically.

#### Building a mitigation plan for the threat

Building a mitigation plan involves the selection of the most effective combination of countermeasures from the list of all the proposed countermeasures for the threat. The decision which of proposed countermeasures will be included in the actual mitigation plan is done by the analyst according to his/her expertise and experience. She will decide upon the most effective group of countermeasures for mitigating the threat - the threat's mitigation plan.

Constructing a mitigation plan is a matter for experts and no simple rule for finding the correct combination of countermeasures can be given. Moreover, the actual mitigation provided by all countermeasures in the mitigation plan is not necessarily equal to the sum of the individual mitigation provided by each of the countermeasures. In order to assist with the computation of countermeasures' contribution to mitigating the threat's risk, the analyst is asked to a) estimate the mitigation level each countermeasure provides to the threat as if it is the only countermeasure in the mitigation plan and b) estimate the total level of mitigation provided to the threat's risk by all countermeasures in the mitigation plan.

#### Identifying relevant attacker types

Associating threats with their relevant attacker types may be helpful in re-assessing the threat's scenario and probability, the exploited vulnerabilities and the potentially damaged assets. All the threat's elements should fit with the attackers' profiles - their qualifications, motivations and accessibility to resources.

#### Specifying tags

Associating threats with their relevant system's tags may be helpful for validating the mapping of the threat scenario on top of the design documents used by the analyst. The tags properties may be used later for viewing risk and mitigation statistics grouped by specific tags e.g. system areas.

#### Identifying relevant entry points

Associating threats with their relevant entry points may be done in correlation with the identification of vulnerabilities exploited by the threat and the attacker types and the tags associated with the threat. Each of the three last steps should be used by the analyst to validate the threat scenario and to help in evaluating the threat's risk and the effectiveness of the proposed countermeasures.

#### **Learning the results**

The following are the outcomes of the threat analysis process:

- List of the system's threats sorted by their risk
- List of the system's threats sorted by the financial damage they cause
- List of individual countermeasures sorted by their overall risk mitigation effect
- List of countermeasures sorted by their cost effectiveness
- Maximal financial risk caused to each asset by existing threats
- Maximal financial risk caused to each asset by existing threats after all mitigation plans are implemented
- Maximal financial risk caused to each asset by existing threats after partial implementation of mitigation plans (use the 'already implemented' flag in countermeasures)
- Total financial risk including all assets
- Total financial risk after all mitigation plans are implemented
- Total financial risk after partial implementation of mitigation plans

Reviewing these results can help the analyst in improving the threat model and in refining the parameters of the entities. It is most productive to check how the model behaves in response to changes in the input data and running various "what if" scenarios since this provides additional insight of the systems' realities.

[1][RiskWatch](#), [Callio Secura](#), [COBRA](#)

[2][CRAMM](#)

[3][Microsoft Threat Modeling Tool](#)

[4][Attack Trees](#) by Bruce Schneier

[5][STRIDE and DREAD](#)

© PTA Technologies 2005

[www.ptatechnologies.com](http://www.ptatechnologies.com)

+972 3 5443085