

Janak J. Parekh

janak@cs.columbia.edu
<http://www.cs.columbia.edu/~janak>

OBJECTIVE

To create solutions. To solve problems. To build new tools. To teach ideas. And, most importantly, to make a difference in computing.

ACADEMIC EXPERIENCE

- Ph.D. Graduate School of Arts and Sciences, Columbia University, New York, NY* 2001-
Expected defense in fall 2006. Anticipated thesis title: "Privacy-Preserving Present
Distributed Event Correlation". Research focus on reliable distributed software
systems, network security, and intrusion detection.
- M.S. School of Engineering and Applied Science, Columbia University* 1999-2001
Emphasis on research in software tools, collaborative environments, run-time
event monitoring infrastructures. Strong academic standing (3.82 GPA).
- B.S. School of Engineering and Applied Science, Columbia University* 1995-1999
Emphases on systems and intelligent systems. Strong academic standing (3.73
GPA cumulative, 3.82 GPA in major; Dean's List every semester).

WORK EXPERIENCE

- Intrusion Detection Systems Lab, Columbia University, New York, NY 2003-
*Lab member, various positions including System Administrator, Project Lead, Graduate Present
Research Assistant*
Engaged in collaborative intrusion detection research, culminating in the *Worminator*
project (<http://worminator.cs.columbia.edu>), an infrastructure for privacy-preserving
intrusion alert sharing. Administered 5+ servers running Linux and Windows.
- Department of Computer Science, Columbia University, New York, NY 2001-2006
Preceptor
Taught six three-credit undergraduate courses as a graduate instructor (preceptor); topics
included Introduction to Computer Science, Data Structures, Discrete Math, and Software
Engineering, with class sizes up to 92 students. Received high evaluation scores (average
4.5/5.0).
- HydraWEB Technologies, Inc., New York, NY 2000-2001
Software Developer
Developed additions and modified existing infrastructure of the company's primary
product, HydraOS, used in their commercial load-balancers. Low-level systems program-
ming in BSDi UNIX and Linux, protocol implementation, and maintenance. Coordinated
software, hardware design and infrastructure with the engineering team.
- Altiverse Systems, Inc., New York, NY 1999-2000
Co-Founder, Technical Lead, System Administrator
Cofounded Internet startup specializing in distributed 3D Internet technologies. Played
major role in product development and product specifications, particularly server-side
network infrastructure, tools and products to realize the company's goals. Assisted in
development of the business plan, hiring and employee coordination, and office systems
maintenance.

Programming Systems Lab, Columbia University, New York, NY	1997-
<i>Lab member, various positions including System Administrator, Project Lead, Graduate Research Assistant, and Project Student</i>	Present

Engaged in a variety of research, primarily involving collaboration tools and event infrastructures. Primary research involved XUES, an event capture and event recognition framework for heterogeneous network/device management and monitoring. Administered 10+ workstations and 5+ servers running Windows, Linux, and Solaris.

Department of Computer Science, Columbia University, New York, NY	1996-2001
<i>Teaching Assistant, Recitation Leader</i>	

Teaching assistant (TA) and recitation leader for 8 semesters in a broad variety of topics, including Intro to CS, Data Structures, Discrete Math, and Graph Theory.

Jantek Computers, Manhasset, NY	1982-
<i>Various positions, including Chief Engineer</i>	Present

Specialize in end-user and corporate IT consulting, dealing with PCs, PC networks, and software development. Involved in long-term projects and relationships on a personal basis with many customers, including small-to-medium organizations in distribution, retail, labor, and manufacturing. Professional instruction for individuals and corporate groups.

OTHER SKILLS

<i>Computing and Software Development</i>	<ul style="list-style-type: none"> • Extensive background in operating systems, networks, Internet platforms, databases. Familiarity with AI, UI, graph theory, algorithm analysis, computer architecture. • Experience in large-scale development in C and Java (incl. JSP/servlets, JMS) • Familiarity with other languages: C++, Python, LISP, Visual Basic, SQL, assembly (MIPS, ARM), HTML, XML, bash, others
---	---

<i>IT Admin</i>	<ul style="list-style-type: none"> • <i>Operating systems/platforms:</i> Windows (32-bit, 16-bit, CE), UNIX (Solaris, SCO Xenix/Unix, Linux, BSD, Mac OS X), NetWare, others • <i>Networks:</i> Windows Active Directory, NetWare, UNIX, LAN/WAN hardware • <i>Internet services:</i> Apache, IIS, qmail, exim, Microsoft Exchange, PHP, Tomcat, MySQL, PostgreSQL, TCP/IP services (incl. BIND/DNS, NAT, DHCP, firewalling)
-----------------	---

SELECTED METADATA

<i>Certifications</i>	Microsoft Certified Professional	1996-
	Novell Certified Network Engineer	1994-

<i>Awards</i>	Microsoft Most Valuable Professional	2005-2006
	SEAS Teaching Assistant Awards (3 semesters in a row)	2000-2001
	Columbia CS Theodore Bashkow undergraduate research award	1999
	Honorable Mention, Computing Research Association, Outstanding Undergraduate Award	1998

<i>Affiliations</i>	Member, Association of Computing Machinery, national organization	1997-
	Member, Tau Beta Pi (engineering honor society)	1997-
	Secretary, Graduate student advisor, Programming contest coach	1997-2006
	Association of Computing Machinery, Columbia chapter	

<i>Hobbies</i>	Handheld computing, baseball, classical music
----------------	---

REFERENCES (OTHERS AVAILABLE UPON REQUEST)

Gail E. Kaiser	Salvatore J. Stolfo
Professor, Computer Science	Professor, Computer Science
kaiser@cs.columbia.edu	sal@cs.columbia.edu

RESEARCH

Ph.D. My thesis is tentatively titled “Privacy-Preserving Distributed Event Correlation”. I aim to address the fundamental problem of inter-organizational information sharing: while many network-based applications can benefit from sharing and correlating information between organizations, people are reluctant to do so for fear of releasing sensitive data.

In particular, I have built an architecture called *Worminator* (<http://worminator.cs.columbia.edu>), whose goal is to share information between organizations to determine common network security threats and to enable preemptive action on those threats. Worminator can leverage different Intrusion Detection sensors, perform privacy-preserving transformations on data, and exchange them in a form that makes it difficult (if not impossible) to trace the source or potentially confidential information—while effectively correlating data to draw conclusions about the threat risk of Internet attacks. Worminator extends prior work of mine on distributed event-based software monitoring, and can be used with a heterogeneous set of sensors and network topologies.

As part of my Ph.D., I was also a research mentor to more than a dozen undergraduate and MS students that did various projects in the Programming Systems Lab. Research varied from XML-based event processing and semantic analysis, to event processing integration with SMTP and IM networks, to porting our technologies on mobile devices. I was also instrumental in the writing and execution of grant proposals from DARPA, NSF, Department of Homeland Security, Army Research Office, and others.

M.S. My Master’s degree work focused on Worminator’s predecessor, known as XUES (XML Universal Event Service; <http://www.psl.cs.columbia.edu/xues>). Originally intended to support “latecomers” to a development-oriented collaborative environment, it became a toolkit for processing events in distributed software monitoring frameworks.

B.S. I first got involved in research with Prof. Gail Kaiser in my junior year in 1997; my bachelor’s thesis work, TaskWeb, was designed to add a hypermedia/hyperlinking environment to PalmOS-based PDAs. My undergraduate research activities made me a finalist for the Computing Research Association’s undergraduate research award.

PUBLICATIONS (*published/submitted*)

1. Parekh, J.J., Wang, K., and Stolfo, S.J. *Privacy-Preserving Payload-Based Correlation for Accurate Malicious Traffic Detection*. to appear in *Large-Scale Attack Detection, Workshop at SIGCOMM*. 2006. Pisa, Italy.
2. Wang, K., Parekh, J.J., and Stolfo, S.J. *Anagram: A Content Anomaly Detector Resistant to Mimicry Attack*. to appear in *Recent Advances in Intrusion Detection*. 2006. Hamburg, Germany.
3. Parekh, J.J., Kaiser, G., Gross, P., and Valetto, G., *Retrofitting Autonomic Capabilities onto Legacy Systems*. *Journal on Cluster Computing*, 2006. **9**(2): p. 141-159.
4. Cretu, G., Parekh, J.J., Wang, K., and Stolfo, S.J. *Intrusion and Anomaly Detection Model Exchange for Mobile Ad-Hoc Networks*. in *IEEE Consumer Communications and Networking Conference*. 2006. Las Vegas, NV.
5. Parekh, J.J. *Worminator (poster)*. in *Symposium on Recent Advances in Intrusion Detection*. 2005. Seattle, WA.
6. Locasto, M.E., Parekh, J.J., Keromytis, A.D., and Stolfo, S.J. *Towards Collaborative Security and P2P Intrusion Detection*. in *IEEE Information Assurance Workshop*. 2005. West Point, NY.
7. Gross, P., Parekh, J.J., and Kaiser, G. *Secure “Selecticast” for Collaborative Intrusion Detection Systems*. in *International Workshop on Distributed Event-Based Systems*. 2004. Edinburgh, UK.
8. Keromytis, A.D., Parekh, J.J., Gross, P., Kaiser, G., Misra, V., Nieh, J., Rubenstein, D., and Stolfo, S.J. *A Holistic Approach to Service Survivability*. in *ACM Workshop on Survivable and Self-*

- Regenerative Systems*. 2003. Fairfax, VA.
9. Kaiser, G., Parekh, J.J., Gross, P., and Valetto, G. *Kinesthetics eXtreme: An External Infrastructure for Monitoring Distributed Legacy Systems*. in *Autonomic Computing Workshop*. 2003. Seattle, WA.
 10. Kaiser, G., Gross, P., Kc, G.S., Parekh, J.J., and Valetto, G. *An Approach to Autonomizing Legacy Systems*. in *Workshop on Self-Healing, Adaptive and Self-MANaged Systems*. 2002. New York, NY.
 11. Gross, P., Gupta, S., Kaiser, G., Kc, G.S., and Parekh, J.J. *An Active Events Model for Systems Monitoring*. in *Working Conference on Complex and Dynamic Systems Architectures*. 2001. Brisbane, Australia.

TEACHING		
<i>Semester</i>	<i>Course</i>	<i>Enrollment</i>
Fall 2001	<i>COMS W3156, Software Engineering*</i> Taught required third-semester course for CS majors. Designed for upper-level undergraduate and masters students, with a focus on skills relevant for large-scale software development including teamwork, software process, and common software tools. Co-designed curriculum with instructor for other section; oversaw group creation, created homeworks and large-scale project assignments.	30
Fall 2003	<i>COMS W3134, Data Structures</i> Taught second of two required courses for many Engineering and Biomedical Informatics non-majors. Designed syllabus, homeworks, and exams. Received a 4.66/5.00 overall instructor rating.	80
Spring 2004	<i>COMS W1003, Introduction to Computer Science in C</i> <i>COMS W1004, Introduction to Computer Science in Java</i> Taught pair of “tandem” courses as a required first-semester course for Engineering non-majors. Modeled syllabus after Prof. Aho’s course in the prior semester, which featured a greater emphasis on Computer Science theory than previous versions of the courses. Taught one combined “theory” lecture per week and coordinated graduate TAs’ “language” labs to correspond to theory material. Designed labs, homeworks and exams. Received a 4.37/5.00 overall instructor rating despite experimental course structure.	22 41
Summer 2004	<i>COMS W3203, Discrete Math*</i> Taught standard discrete math course in the summer session.	4
Fall 2004	<i>COMS W3134, Data Structures (non-majors)</i> Similar to the course taught in fall of 2003; received a 4.58/5.00 overall instructor rating.	58
Spring 2005	<i>COMS W1004, Introduction to Computer Science in Java (majors and non-majors)</i> Redesigned syllabus for and taught yet another new version of COMS W1004, now designed as a course required for Engineering non-majors and an “introductory” course for majors (to be followed with COMS W1007 for majors). Taught two lectures per week, blending both theory and programming practice. Received a 4.38/5.00 overall instructor rating.	92

**Unfortunately, the Engineering school changed the evaluation system between 2001 and 2003, and the old evaluation scores have been lost. Additionally, the summer session does not use the same evaluation system as the Engineering school.*

COMMUNITY ACTIVITIES

ACM JETT coordinator and instructor: Helped to launch Java Education for Teacher Training (<http://jett.acm.org>), a program to help high-school teachers learn Java skills in preparation for the upcoming programming language change in the AP Computer Science high-school exam. Under the supervision of Professor Betsy Sklar and in tandem with colleagues in the national Association of Computing Machinery chapter and the national Advanced Placement (AP) chapter, assisted in the organization of two pilot workshops at Columbia University representing high schools from the NYC metropolitan area in May '03 and Feb. '04. Taught high-school teachers a variety of Java programming skills, including graphical user interfaces and data structures.

2002-2004

Computer Science Facilities Committee, secretary: Took minutes for departmental IT oversight committee. Made suggestions to help IT infrastructure development for both students and faculty.

2001-2006

TA Room organizer, maintainer: Worked with other students in 1998 to petition the Engineering school to allocate dedicated teaching assistant space for Computer Science. Previously, CS TAs held their office hours in crowded AcIS computing labs—a significant source of distraction for other students. After the successful acquisition of space on the first floor of Mudd in the fall of 1999, assumed the role of TA room maintainer for several years, primarily involving the acquisition and management of computing equipment. The department took over full-time management in 2003.

1998-2003

ACM Programming Contest coach, regional contest host: Organized and coached teams of Columbia students to compete for a slot at the ACM International Programming Contest (<http://acmicpc.org>). Teams consistently ranked amongst the top teams in the Greater New York region. Additionally, hosted the regional ACM programming contest (<http://www.acmgnyr.org>) at Columbia in fall 2002; co-organized the contest and computer systems to support 54 teams and over 160 participants from schools as far away as Cornell.

1999-2005

ACM Columbia chapter secretary, graduate student advisor: In 1998, joined a few other undergraduates to resurrect Columbia's ACM chapter (<http://acm.cs.columbia.edu>)—a student preprofessional organization dedicated to improving the reach of Computer Science—after two decades of chapter inactivity. Served as secretary in the 1998-1999 academic year. Helped organize numerous academic and social events, including teaching sessions, contests, speakers, and review sessions over the years. Currently graduate student advisor. With another graduate student, created the annual Computer Science Research Fair in 2000, giving undergraduates an opportunity to tour and talk to various research groups and increase their opportunities for research.

1998-2006